UNIVERSITY OF DURHAM

Department of Computing

PROCEEDINGS OF

NETWORKSHOP 6

9 and 10 APRIL 1980

CONTENTS

PAPERS NOT RECEIVED

Papers were not received on the following items :-

1) Job Transfer and Manipulation Protocol

2) Planning the Transition to standard networking in London

INTRODUCTION

The sixth University Networkshop was held in the University of Durham from 9th to 11th of April 1980.

As usual, all speakers have been asked, coaxed, or bullied into providing some form of synopsis of their talks and these constitute the proceedings that follow.

Following suggestions at Canterbury for shorter workshops , it was found possible to squeeze the program into a day and a half with some group working sessions on the evening of arrival. This seemed to prove successful on this occasion at least.

Most Universities sent at least one representative, as did several of the Polytechnics and Science Research Council Laboratories. Future workshops will drop the theoretical limit of one representative per institution and will regularise the de facto situation of allowing additional representatives within the limits of accommodation available.

As usual the major part of the workshop was concerned with bringing us up to date with developments in communications. Nothing fundamentally new emerged although, in this age of materialism, it is comforting to report that there are still a few souls with the pure faith to continue their belief in PSS DAY1.

Everybody (or nearly everybody) agrees that standard communication protocols and standard implementations of these in hardware and software are highly desirable goals. Standard protocols are emerging for many aspects of current needs (although not all have been greeted enthusiastically).

There is also strong interest in high bandwidth local area networks involving techniques such as rings or ethers. Many of us envy the few institutions that have working systems in this area - possibly without giving full weight to the amount of sweat and tears behind these implementations.

The general advice is still to be patient and wait until all the standard hardware and software is available, preferably from the manufacturers. This is excellent advice in theory but the half-yearly networkshops flash by and commercial availability stays, if not as far off, at least still a good distance in the future. I would sum up my feelings with the questions - how long can we afford to wait and what do we do in the interim? Perhaps the statement by Daresbury that they are to write their new network-manager in a mixture of macro-assembler and FORTRAN, typifies the likely answer.

I would like to record my thanks to Barry Charles of the JNT who did most of the preparation of the workshop programme and to Paul Jones of Durham who did everything else (including the preparation of these proceedings).

The next workshop in the series will be at the University of Oxford in September 1980.

A.A. Young
University of Durham
July 1980

HIGH-LEVEL PROTOCOL TESTING AND VERIFICATION

Keith Bartlett

Data Communications Protocol Unit

High Level Protocol Testing and Certification

A discussion session on the certification of high level protocols was attended by approximately 25 people. The basis of the discussion was a development programme at NPL Teddington designed to create testing procedures for high level protocols. These tests would check implementations of the protocols for correcteness and performance. The actual testing would almost certainly, not be carried out by NPL and discussions were currently taking place with a view to NCC being the actual testing body. The envisaged test centre would operate along commercial lines. Such a centre could eventually become a certification centre when (and if) protocol implementations required certificates of correctness before purchase or use.

There was support for the idea of certification because it allowed managers of sites and services to connect to external facilities with confidence. In spite of this, some reservations about the credibility of a certification centre and manufacturer acceptance of it were expressed. No such reservations were evident, however, concerning a development aid which would enable implementors to check out and ruggedise their products before marketing. This is a natural by-product of a certification centre. In view of the emphasis on this latter facility, the term 'Protocol Assessment Service' rather than 'Certification Centre' came into use during the discussions. The development aid part of this service was seen to be an urgent requirement with at least one manufacturer needing such a facility in the summer of 1980.

A quick poll suggested that the Yellow Book (Transport Service), Blue Book (File Transfer) and Red Book (Job Transfer and Manipulation) protocols would create a demand amounting to approximately 20 implementations requiring service from the centre in the year 1981 - this service being a combination of development aid and more formal checking.

The NPL programme envisaged the early development of 'quick and dirty' tests with these and the associated experience being used as the basis of more fully developed tests towards the end of the project. Initially testing techniques for the Yellow and Blue book protocols were being devised but others would follow as required. The development programme was a fairly comprehensive one estimated to take 2 years to complete. This was thought to be a long time, particularly so given the stated demand for the development aid at least,in the current year.

REPORT FROM THE JOINT NETWORK TEAM

Roland Rosner

Joint Network Team

REPORT FROM THE JOINT NETWORK TEAM

The Joint Network Team is now four-strong with a fifth recruit joining in mid-May and the sixth post still to be filled. The team's work may be broadly classified into policy, planning, the development of components and advising the funding bodies on communications topics.

The formulation of networking policy is concerned principally with mechanisms for creating common communications arrangements to serve the needs of all the funding bodies. Among the major areas being considered are the financial implications of the various schemes proposed as well as the problems of managing complex network hierarchies. Planning deals with the transition in particular cases from present data transmission methods towards more flexible networking within the broad guidelines of the overall policy.

Among the team's more readily visible activities are a number of projects in progress under JNT contracts to produce widely applicable software and hardware packages for constructing networks and attaching subscribers' equipment to them.

As a specialist extension of the Computer Board Secretariat, the team currently has to advise the Board on the communications issues arising from any university submission.

Internally, the team is organised so that each member has a set of specific technical and managerial responsibilities and also acts as the contact for communications activities and plans in several geographic regions and Research Councils. During the build-up of the team, the assignment of tasks may vary and therefore no names are published in these proceedings. Rather than giving an exhaustive list of all the JNT's commitments, we single out the items of most interest.

The SRC is undertaking a major review of its computing provision and the JNT is being asked to contribute towards a submission on communications as part of this exercise. A paper has already been drafted making recommendations on the future growth of the SRC/NERC network.

Machine range products already contracted or under negotiation include the implementation of X25, XXX, the Transport Service and the File Transfer Protocol

on the following machines:

DEC10 (TOPS-10), VAX, PDP-11 (RT-11 and UNIX) and ICL 1900 (GEORGE 3).

More details are given in the summaries of the machine range sessions. Mike Sayers (Hatfield Polytechnic) is continuing to act as consultant to the JNT on DEC10 networking.

There is a project at the University College of Swansea to develop a highly programmable terminal contention/switching facility based on an LSI-11. More substantive discussion of the general terminal problem appears in the talks by Peter Higginson (UCL) and Rick Blake (Essex) who is acting as a consultant to the JNT on this topic.

A report on the status of PSS and a request for funds to cover the first year's usage have been submitted to the Computer Board. It would appear sensible for the JNT to act as the focal point for interactions with the Post Office and, in this capacity, the JNT will circulate a list of recommended subscription-time options to prospective PSS subscribers. John Thomas (SWURCC) is assisting the JNT in a consultancy capacity on PSS, private switches and related topics.

Local area networks continue to occupy a considerable fraction of the JNT's time. However, without prejudice to the conclusions of the LAN session later in the proceedings, it is clear that ring technology has not yet reached the stage where standards can be defined. The considered view of the experts is that it would be unwise for rings to proliferate in an uncontrolled fashion until the long list of problems have been tackled and more experience has been collected from a very limited number of pilot projects. Ray Chisholm (Edinburgh) is acting as a JNT consultant on local area networks.

Following discussions at the last Networkshop, the JNT has devoted much attention to the handling of contracts including the drafting of specifications, the monitoring of progress, the maintenance of milestones and timescales and the staging of payments. Not surprisingly, the proper performance of these activities entails significant effort by JNT members.

Roland Rosner
Joint Network Team

STATUS REPORT ON PSS

Pat Morrison

Post Office

Telecommunications Headquarters

Marketing Executive

Business Systems Department

Telematics Division

Seal House
107-108 Upper Thames Street
LONDON
EC4R 3TH

Telephone:
01-357 4361

Telex: 883051

Date
May 1980

In any reply please quote:

Your reference:

Post Office Telecommunications

Dear Sir

At the recent User Forum, a statement was promised giving the latest position on the Development Aid. It is regretted that the Aid will not now be available until such time that it can support multi-access at Level 2 and an intermediate Level 3 to PSS specification. This is expected to be the beginning of October. Multi-access will be provided upto 9,600 bit/s and the Level 3 will be gradually enhanced to cover all PSS facilities by the end of December 1980.

It is recognised that this will introduce severe problems for our customers but this course of action is preferred to the alternative of offering the Development Aid with very limited access and with limited facilities.

The major interest in the Development Aid is associated with Permission to Connect to PSS; although it has been stated that use of the Development Aid is not mandatory for seeking Permission to Connect it is, of course, recognised as being highly desirable. It is therefore suggested that if you have an implementation of an interface to PSS, please inform Vince Taylor or Alan Jones in the Attachments Section of the Post Office. You will then be informed of the possible charges which apply, and, following your agreement to proceed, you will receive a questionnaire. If the Attachments Section are confident that your implementation is acceptable, you will be granted temporary permission to connect to PSS for a period of time prior to granting full Permission to Connect.

At the User Forum a question was raised regarding a possible 'free period of use of PSS' when the service opens. Recognising that PSS is a new service, a short 'bedding down' period may be required. During this bedding down or pre-operational phase the service could be subject to interruptions, some of which may be prolonged. In recognition of this no PSS charges will be raised during this phase.

The inter-connection of PSS to IPSS is planned for early 1981, at which time the transfer of customers from IPSS to PSS is expected to commence; existing users of IPSS, who have already been advised of our inter-connection plans, will be notified individually of the date of transfer to PSS. There is still no date that can be quoted for the inter-connection of PSS and Euronet.

If you have any queries, or would like to discuss your requirements in detail, please call Mrs Mandy Pritchett on 01-357 4181 or Des Mills on 01-357 4361, or write to us at the above address. The address of the Attachments Section is:-

Post Office Telecommunications
Tenter House
45 Moorfields
LONDON
EC2Y 9TH

Yours faithfully

G Dale

G DALE
Head of Telematics Division

STATUS REPORTS FROM EARLY PSS SUBSCRIBERS

Ian Service

University of York

Status report on connecting DECsystem-10's to PSS

The DECsystem-10 project is well under way and it is anticipated that the DECsystem-10 will be ready to connect to PSS on day one.

Implementation:

Level 2 — The level 2 protocol is now HDLC, as defined by the PSS user guide. This has been tested to independent implementations at the Daresbury and Rutherford Laboratories. This protocol is now in production use to SRCNET.

Level 3 — The level 3 protocol is based on the implementation for SRCNET and this has now been converted to be PSS compatible. Back to back testing is now under way.

Transport Service

Transport Service addressing is implemented and the rest will be implemented as required.

Terminal Protocol

The SRCNET ITP is in production use and an implementation of "XXX" is defined for PSS is under way.

FTP — A spooler which will run on the DEC-10 has been designed and is now being coded.

A line to the local PSS exchange has been installed by the Post Office and modems have been delivered and tested. Permission to connect has been applied for and the Post Office are expected to reply shortly.

Further details on the DECsystem-10 project are given in the paper "Packet Switching and the DECsystem-10" elsewhere in these proceedings.

J D Service
University of York

UNIVERSITY OF YORK

DEPARTMENT OF COMPUTER SCIENCE

Packet Switching and the DECsystem-10

Edition 1.1

February 1980

Author: J D Service

There is no standard DEC software for connecting a DECsystem-10 to a packet switched network, but work has been under way in the British university community to produce a connection. This paper describes some of that work.

J D Service, Department of Computer Science, University of York, Heslington, York YO1 5DD. Telephone: 0904 59861 Telex: 57933

Introduction

This paper describes a series of projects which have been undertaken at the University of York which have the aim of integrating the DECsystem-10 into an X25 networking environment. With the benefit of hindsight these essentially separate projects are described as though they were all steps on the way to producing a single X25 package for the DECsystem-10.

History

The first stages of the X25 development were undertaken at the Hatfield Polytechnic where the author was working on a project to implement ANF-10 on a non-DEC computer. At that time the Polytechnic was connected to the British Post Office's Experimental Packet Switched Service (EPSS), and an unsatisfactory asynchronous terminal link existed between the DECsystem-10 and the EPSS link machine (another non DEC computer).

One Friday afternoon coffee break in 1977 a project was dreamed up which made use of the combined ANF-10 and EPSS expertise to design a mapping between ANF-10 protocols and EPSS protocols. This `midnight hack', though not using X25 protocols, was the real starting point of all the X25 work described in this paper.

During 1978 a study was undertaken by Dr M D Sayers of Hatfield Polytechnic with the aim of recommending a suitable method to connect the two networks of the British Science Research Council (SRC). These were

- the ANF-10 network of the Interactive Computing Facility (ICF) with major DECsystem-10 nodes at Edinburgh and Manchester,
- the main X25 network, SRCNET, with major IBM nodes at Rutherford and Daresbury.

The proposed aim of the connection of the two networks was to provide remote terminal access to the DECsystem-10's from SRCNET, and job submission to the IBM machines for the DECsystem-10 users.

Alternatives for connecting a DECsystem-10 to an X25 network

A. Replace ANF-10 by X25 as the primary DECsystem-10 network protocol.

B. Implement X25 on the DECsystem-10 in addition to ANF-10, possibly as a separate front end.

C. Provide a black box to translate between ANF-10 and X25 - a gateway.

All three options received very serious consideration and though C was eventually chosen it is worth discussing why the other two were rejected.

The idea of replacing ANF-10 by X25 was very attractive, and it is almost certain that suitable higher level protocols could have been found to give the same performance as the current ANF-10 terminal protocols, and the other device protocols would not have proved difficult. However this would have been a major undertaking and when the problems of support were considered it was obvious that this was not practical as a user project.

Implementing a separate X25 front-end was very nearly the chosen solution especially as a DL-10 based front-end, supporting IBM 2780 protocols, was already in use and it seemed likely that this could be adapted to support X25 protocols. While this solution would have suited the particular requirements of the ICF machines, since they were both KI processors and would therefore support DL-10's, it was pointed out that this solution could not be applied to the newer KL processors (especially the 1091's) that were by then being installed in British universities. In fact the main reasons for rejecting this solution lay outside the original requirements, but it seemed desirable to be able to provide a common solution for all the DECsystem-10's in the academic community.

The chosen solution, that of a gateway between ANF-10 and SRCNET, was suitably isolated from the main system to ease the support problems and sufficiently hardware independent to remove the restriction regarding the DECsystem-10 processor type. Thus in late 1978 a proposal by the University of York (where the author was now working) was accepted and a project was funded, jointly by the SRC and the Computer Board of the British universities, to implement a gateway between ANF-10 and an X25 network (in the first instance SRCNET).

Specification of the Gateway

The ANF-10/SRCNET gateway was required to support the terminal protocols used by both networks and to provide an interface such that a terminal connected to either network could access hosts on the other network. In addition a block mode transfer device was specified to enable blocks of unformatted data to be transferred between an ANF-10 host and a foreign host, in effect an X25 level 3 interface for the DECsystem-10. This original specification did not cover such things as File Transfer and Transport Service, nor did it require implementation of the CCITT high level protocols.

Design of the Gateway

When one considers most modern networks they appear to have very similar structures because they are all composed of several layers of protocols, and if one is considering translating between two networks, it appears as though one could perform the translation at several alternative levels.

Consider the three networks:-

		ANF-10	SRCNET	CCITT X25	
Level 2		DDCMP	BSC (HDLC)	HDLC	
Level 3		NCL	X25 (subset)	X25	
Level 4	(TTY)	NCL DAP-TTY	EPSS-ITP	X3/X28/X29	for terminals
	(other)	NCL DAP-devices	FTB-B	?	for files

Figure 1

This figure shows the relationship between the different protocols of each network and shows the alternatives for protocol translation.

A level 2 mapping between HDLC and DDCMP would be possible but would simply be a replacement for DDCMP, not a real network translation.

Level 3, NCL and X25 level 3, is a more serious thought and in fact the previously mentioned Hatfield project for a gateway to EPSS attempted to use this approach. In X25 terms this could involve treating the whole X25 network as though it were a single ANF-10 node and converting thus:-

X25 RESTART sequence	< - >	NCL START/NODE/NEIGHBOURS sequence
X25 CONNECT	< - >	NCL CONNECT
X25 DISCONNECT	< - >	NCL DISCONNECT
X25 logical channel number	< - >	NCL link address

and in fact this conversion appears quite straightforward. Unfortunately some X25 control messages do not map directly onto NCL messages, for example RESET, and since the higher levels of protocol are not compatible these have to be translated or implemented at some other point. Although the Hatfield experiment did work it contained so many modifications to the idea of a level 3 mapping that it was in effect a combined mapping of levels 3 and 4 of EPSS.

The Hatfield gateway did, however, point the way to what should have been obvious from other researchers' papers, that the obvious place to perform a mapping between networks is at the data transfer level, i.e. the transport service level. This allows each network to treat the other network as though it were a device, and so an ANF-10 to X25 gateway has the structure shown in Figure 2.

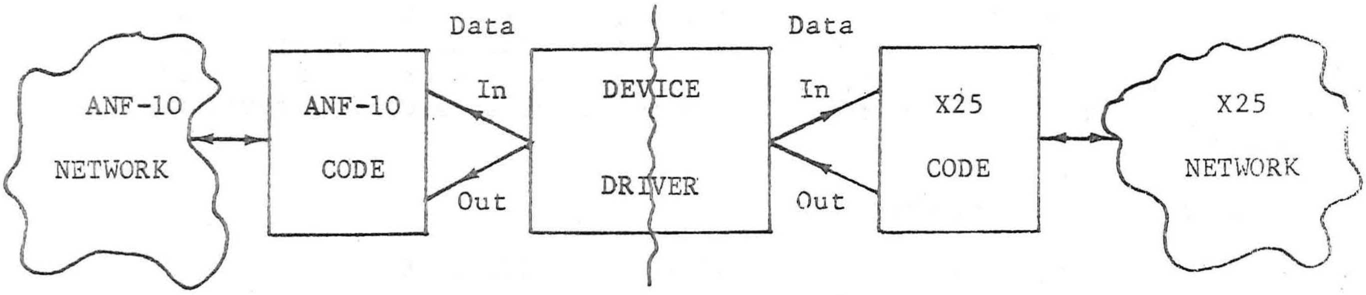


Figure 2

It is worth pointing out that what is being described is almost the textbook implementation of a gateway, that is a transport level gateway, with different terminology. This fact is not obvious because ANF-10 and current X25 high level protocols do not use a discrete transport service. Instead each protocol has its own transport layer built in.

Structure of the Gateway

Having decided how the gateway would work, the next stage is to design a software package which would be straightforward to implement. Once again various options were considered, including basing the software on a non-DEC ANF-10 node, but the chosen solution was to use a PDP-11, in fact a DN80/DN200 series node and modify the standard DEC code.

In theory any DN80 or DN200 series node would do but the actual implementation has been done using:-

DN200 (i.e. a PDP-11/34)
32 KW of core
a DMC-11, for the ANF-10 connection,
a DUP-11, for the X25 connection
a console decwriter, for logging and general usage.

NOTE

> This is the configuration for SRCNET, a private network with no access charges. The configuration for a gateway to access the public network has been extended to include a small disk.

As this system already comes with complete ANF-10 software there was no need to develop anything for that side of the gateway. The first step was to design a piece of software which implemented levels 2 and 3 of the X25 network and which followed the style of the DEC code, so both software packages could use many of the same routines. This gave us free-store management, with all of free store being held as a linked chain of chunks; a queue structure which is implemented in macros; and many other routines which would have been very tedious to implement. It should be pointed out at this stage that the tasking system which is supplied with the DN80/DN200 code (DNTASK) is not used and all of the X25 software is integrated into the in-line code. In fact the modifications simply comprise of a few calls to the X25 code inserted at loop level and clock level.

Having now both sides of the gateway the next stage is to implement the cross-over routines. At first sight this may appear trivial but there are problems:-

A. How does Call set-up work; how do we communicate the X25 address over the ANF-10 network to the X25 software.

B. How do Call Clearing and other exception conditions get communicated to the user, in both directions.

C. How does the X25 user establish ANF-10 connections once he has been connected.

This is where a proper transport service on both networks would have simplified matters but this was not available, neither at that time was the British Transport Service specification. For this reason two alternatives were discussed:-

- use the TSK device in the DECsystem-10 and carry either the whole of the level 3 packet in a NCLTSK message using a free running program to build X25 control and data messages, or interface the NCLTSK set-up and clearing to be X25 set-up and clearing and implement only the higher protocols in a free standing program.

- provide a mapping between NCLDAP protocols and X25 high level protocols and do complete protocol conversion in the gateway.

Both of these alternatives have advantages and so we chose to implement both and use the first method, that of the TSK device, for process-to-process communication, and the second for terminal access. The reasons for treating the two modes of access differently are discussed in the next two sections.

This has now given us a gateway whose structure is:-

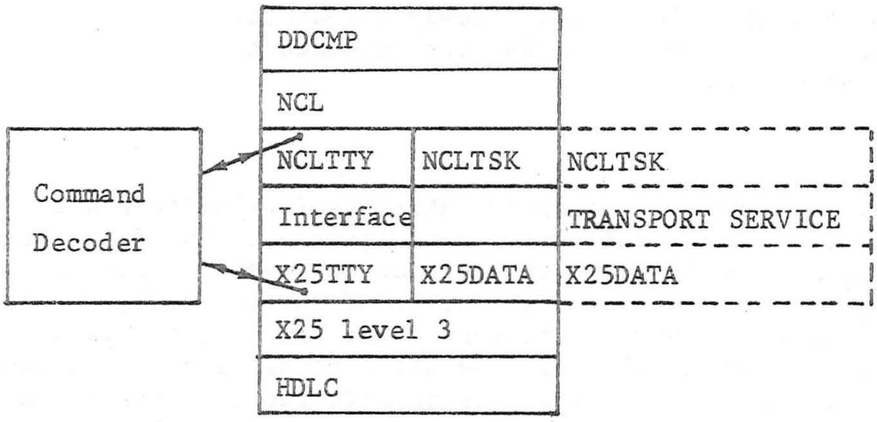


Figure 3

NOTE

The normal device drivers for the ANF-10 devices are not shown but they can, and do, exist in the gateway.

Figure 3 shows two sections which have not been discussed so far; the command decoder which is described in the following section on interactive terminals and the transport service which is in dotted lines and is discussed later in this paper.

Having now discussed the block structure of the gateway it is appropriate to describe how the gateway is used.

Interactive Terminals

It has already been stated that terminals are implemented by mapping between appropriate high level protocols in ANF-10 or X25 (ITP in SRCNET and X3/X28/X29 in the public network), but it has not been explained how call set-ups are propagated between the networks nor how we deal with exception conditions.

In the case of the interactive terminal an obvious solution exists. That is to provide the user with a command level interface and require him to type in the appropriate call set- up parameters for the other network, and this is how it is done in the gateway. The gateway has been extended from being a simple node to being an ANF-10 host with a command decoder (MCR), and all interactive users (from either network) are connected to the command decoder once they have connected to the gateway.

Thus if an ANF-10 user is connected to a DECsystem-10:-

```
.SET HOST GATEWAY
WELCOME to the Gateway
[GATEWAY(16)]CMD> CALL   0500.064700
[GATEWAY(16)] Call Setup
Daresbury TSO system - Enter LOGON or LOGOFF
```

Example 1

Example 1 shows a user creating a call to an IBM system running TSO, where the gateway and foreign host output are underlined. The user interface to the gateway is as similar to that of TOPS-10 as possible, and the gateway performs the intra-line editing that a DECsystem-10 user expects.

An incoming call follows a similar pattern, only he must type "SET HOST" to connect to another DECsystem-10 host. He then appears to that host as though he were any ANF-10 terminal.

The gateway can then provide further facilities for the interactive user, such as HELP, the ability to change terminal parameters, call statistics, and a break-in facility to handle hung calls. In addition, because we can assume a user is present, exception conditions can be reported to him by simply outputting a message on his terminal.

NOTE

> The command decoder on the gateway is very simple and processes single line commands only.

However the two main advantages of this approach are:-

1. It is really a gateway between the ANF-10 terminal network and the X25 network and as such it does not require a DECsystem-10. This means that the external network is available to ANF-10 terminal users even when the DECsystem-10 is not running, and users do not require any expensive DECsystem-10 resources (e.g. a job slot) when they are making an outgoing call.

2. Because the incoming call is seen by the DECsystem-10 as though it were an ANF-10 terminal connected to the gateway node, the user has the same facilities as any other ANF-10 terminal. Any restrictions that exist are imposed by his own network and in particular his local terminal handler. This lack of restriction would not be true if the incoming calls were handled by a process on the DECsystem-10, as its interface to TOPS-10 would then be via a PTY with all the restrictions that brings.

Process-to-Process Interface

We have already said that process-to-process communication uses the TSK: device as its interface to the DECsystem-10 job. This gives block mode transfer through the gateway, as the gateway behaves as an ANF-10 host and implements the `remote´ end of a TSK link. Once the gateway has received data via the TSK link it repackages it as an X25 data message and sends it. Incoming data is treated in a similar fashion.

The Call set-up problem must be solved and in this case there is no user to interact with. An analogous system would be to implement a new protocol, ideally a standard transport service, and carry this protocol as TSK data messages. A running job could then interact with the gateway. The main disadvantage of this approach is that the gateway must police this protocol, and duplicate a large amount of checking that is already done at TSK level.

The alternative is to use the TSK names as X25 addresses and use X25 disconnect reason codes as NCL disconnect codes, and similarly map the other message types. Unfortunately, in TOPS-10 6.03A TSK names are limited to six characters and general control of the TSK device is not very good as the normal OPEN/LOOKUP/ENTER/OUT/IN/CLOSE UUO´s are all that is available to the job. The restriction of TSK names to six characters would be acceptable if the gateway translated names to X25 addresses, and all the other deficiences could be tolerated. Luckily a solution to this problem is around the corner, as there is a new UUO (the TSK.UUO) in the 7 series version of TOPS-10 which is in effect a transport service interface. This UUO allows TSK names of up to one hundred characters for both local and remote processes and provides better conrol of the TSK link. Thus the TSK device is defined to be the DECsystem-10 process interface and in TOPS-10 6.03A restrictions have to be accepted. A separate document (reference 9) describes using TSK´s as a transport service, and describes a format for addressing in the DEC-10.

The SRCNET Gateway

Most of what has already been described was implemented for the SRCNET gateway specified earlier and has been operational for some time; terminals since September 1979 and TSK transfers since November 1979. The software is now in production use and the only changes being made to it are those required to maintain compatability with SRCNET. The SRCNET network is now slowly evolving towards compatability with the British Post Office´s network, and this is described in the next section.

The Post Office Network

The British Post Office are in the process of commissioning a packet switched network based on X25 protocols, and this network (called PSS), is expected to become operational in May 1980. The University community in the U.K. is fairly heavily committed to using the network and several projects have been commissioned to ensure that most types of computers in the community are capable of connecting to PSS. In addition all new computers must have an X25 capability.

There is no DECsystem-10 X25 capability, so the Computer Board of universities (in the guise of the Joint Network Team) financed a new project to convert the SRCNET gateway into a PSS gateway. This contract was also awarded to the University of York and the work is well under way. It is anticipated that it will be possible to connect a DECsystem-10 to PSS in May.

The specification for the PSS gateway is essentially the same as that for the SRCNET gateway but the terminal protocols (among others) are different (X3/X28/X29 instead of ITP). Also the gateway is expected to be capable of handling other higher level protocols adopted by PSS.

Another major difference between SRCNET and PSS is that a PSS user will incur usage charges. Hence the gateway must provide access control and log network usage on a per user basis, neither of which is required for SRCNET.

The access control is being provided by adding a LOGIN command to the gateway and by recording the call statistics to a small disk on the gateway. The call statistics will be copied from the local disk to the mainframe at an off-peak time, and used to re-charge to the appropriate accounts.

File Transfer Protocols

So far very little has been said about file transfer except to mention the process-to-process interface that would be used to carry file transfer. The reason for this is that work on a file transfer system for the DECsystem-10 has been rather slow in getting started. However a project is now under way at the University of York to implement a file transfer spooler which will use the protocol FTP-B.

FTP-B is widely accepted within the U.K. as a standard file transfer protocol and many implementations of it exist. FTP-B is documented in the so called blue book, reference 11, and it is supported by an implementors group.

Early in 1979 the software house CAP undertook to produce a "Functional Specification of FTP-B for the DECsystem-10" (reference 10) and this was completed in May 1979. After this functional specification was completed an implementor was sought but work did not commence until January 1980 and yet another project was started at the University of York.

This FTP-B project implements the file transfer system as a spooler which will be integrated into Galaxy (V4) and which will provide a file transfer queue which appears very similar to that of the other spoolers.

The spooler will include such facilities as:-

multiple simultaneous transfers,
unattended answering of incoming calls,
checkpoints,
translation between different character codes and file formats.

It is anticipated that a preliminary version of the spooler will be available early in the latter half of 1980 with a production version being in use before the end of the year.

The Transport Service

It has already been stated that much of this work was carried out in ignorance of other work on a standard transport service. Now that work is known some account must be taken of it. Already the Transport Service Addressing scheme is a de facto standard and it seems likely that the rest of the 'yellow book' (reference 3) will achieve this status at least within the U.K. Therefore the gateway is being converted to support the transport service as an alternative path through the gateway (see Figure 3) and changes will be made to the DECsystem-10 interface when a transport service specification is an accepted standard. However the final shape of that standard is not sufficiently clear to justify any description of the current state of the DECsystem-10 interface.

Other Standard Protocols

It is obvious that the correct specifications for packet switched networks are not the final ones, especially in the field of high level protocols. Already a new terminal protocol TS29 is being proposed and a Job Transfer Protocol is (hopefully) shortly to be announced.

If these, or any others, become standards it is hoped that the structure of the gateway is sufficiently flexible to accommodate them without too much trouble. The software was designed with this thought in mind.

The Future

Many of the projects described in this paper are continuing into the near future and as the network scene is still evolving it is obvious that many changes will have to be made to the gateway. At this time it is not clear what these changes will be as they are to a great extent dependent on who we wish to communicate with, and what they have chosen to implement.

Summary

This paper was written both as a submission to the meeting of the DECUS Special Interest Group (SIG) on Networks held in Zurich in February 1980, and also as a general description of work being undertaken at the University of York. As such it is not intended to be either a functional description or a technical description of the gateway. If the reader required further information he is referred to the User Guide (reference 6), the Gateway Program Logic Manual (reference 7), A File Transfer System for the DECsystem-10 (reference 8), and the ANF-10 Implementation of the Transport Service (reference 9).

We hope that this paper gives some insight into the history of the gateway and presents some of the arguments for the gateway, though as was said in the beginning, history is always written after the event and only contains what the author thinks appropriate to remember.

Acknowledgements

The author would like to acknowledge his colleagues at the University of York, David L Atkin and Elisabeth Aylmer-Kelly, who have worked on the X25 projects, and also the project consultants, Dr M D Sayers (The Hatfield Polytechnic) and Dr R G Blake (The University of Essex).

References

1. Proposal for the connection of the SRC/NERC network to the ICFC network - Dr M D Sayers - Network Liaison Committee - 29 June 1978.

2. Why a DECnet (ANF-10) Gateway - Dr M D Sayers - 27 July 1978.

3. A Network Independent Transport Service - Post Office Packet Switching Group 3 - 29 January 1980 (DRAFT).

4. CCITT Recommendations X25, X3, X28, X29.

5. PSS Provisional Specification - British Post Office.

6. The DECsystem-10 Network Gateway User's Guide - J D Service - January 1980.

7. SRCNET/ANF-10 Gateway Program Logic Manual - J D Service - January 1980.

8. A File Transfer System for the DECsystem-10 - E Aylmer-Kelly- February 1980.

9. The Realisation of the Network Independent Transport Service on ANF-10 - J D Service - February 1980.

10. A Functional Specification of a File Transfer Protocol for the DECsystem-10 - CAP (London) Ltd. - April 1979.

11. FTP-B A Network Independent File Transfer Protocol - The High Level Protocol Group (British Post Office) - December 1977.

TRANSPORT SERVICE STATUS REPORT

Peter Linington

Data Communications Protocol Unit

Report of the Study Group 3 Working Group on the Transport Service

Since the last User Forum meeting, the Study Group 3 Working Group on the Transport Service has progressed with the revision of the draft previously circulated for comment. The recommendation has now been re-issued, incorporating the comments received and taking note of national and international discussions, and of the practical experience gained from EPSS. The major part of the document is now stable and provides a practical interim standard for the next few years while international standards are being agreed.

The aim of defining a transport service is to define a clear division between the activities of data communication and data processing. Such a division allows the adoption of new communication offerings without invalidating existing investment in existing processing implementations. The aim is to allow the replacement of one network or protocol with another in a way which contains and limits consequent change.

The changes foreseen include the decision to use combinations of dissimilar media, if economics so dictate. An important case is the use of a local private network within an organization in conjunction with a Public Data Network between organizations.

The service referred to in the title of the document is the set of properties and assumptions that can be relied upon by the communication user when designing the way in which his higher level activities will be performed. The definition of a service is, in this context, concerned with the expected properties of the communication. It must be distinguished from the specification of interfaces within a system which allow access to the communication components, and contain many local implementation choices that do not need to be standardized.

The definition of the service is in terms of a set of primitive messages that the user may expect to issue or receive. The available primitives are:

- call establishment (CONNECT)
- call acceptance (ACCEPT)
- call disconnection or rejection (DISCONNECT)
- data transfer (DATA)
- address data transfer (ADDRESS)
- forcing data delivery (PUSH)
- priority data (EXPEDITED)
- resynchronisation (RESET)

The strategy used in the definition is to work within the Reference Models being developed by CCITT and ISO to describe communication. They identify a number of nested services, including a Transport Service and a Network Service. The CCITT

work has lead to the conclusion that the two services exhibit the same set of primitives, behaving in similar ways, but differ in the quality or efficiency of their provision.

We wish to minimize the implementation and operational cost of systems. We therefore stress the advantage of using the available communication medium to its full, exploiting all the facilities whose provision has already been paid for. This involves a choice for each type of network. The aim is to provide a service independent of the details of the available medium, but the way this is achieved must be related to the possible behavior of the medium, and so is necessarily network dependent.

Once this network independent service has been achieved, its quality can be enhanced in a network independent way, using optional network independent protocol modules. An example might be the use of an optional a multiplexing module when a single network connection must be shared. This approach reduces the size and number of different components which must be maintained, without the operational overheads associated with unused facilities.

The recommendation includes description of an addressing mechanism which allows the specification of connections between parties which are loosely co-operating, but which have not been integrated managerially to the extent of co-operating in a single addressing scheme. Such mechanisms are required if organizations which have developed independently are to be able to communicate without internal re-organization.

The technical details are not summarized further here, since they are covered fully in the document itself. The document consists of a main part, which specifies the service, and a set of annexes defining how it can be achieved. Of these, the first deals with X.25 based networks, the second with X.21 networks and leased lines, and the third with a network independent multiplexing function. Of these, the main part and the first annex have been widely circulated for comment, and are now stable. The remainder of the annexes are newer material which may still evolve in the light of further discussion.

TS29/XXX STATUS REPORT

Peter Higginson

University College London

INDRA Working Paper

INDRA Note No 900
28 May 1980

Progress on Character Terminal Protocols

Presentation to Networkshop 6
Durham, 9-11 April 1980

by P.L. Higginson

Abstract

This note is a summary of the presentation at Networkshop 6; it covers the "Green Book" on recommended character protocols for PSS, and a number of implementation and likely future issues.

Department of Computer Science
University College London

22

1. Major Issues in the Green Book

The major issues addressed by the Green Book are:

(i) The recommended way of working for Hosts to access character terminals over PSS.
(ii) Recommended settings for PSS terminal profiles (Annex C).
(iii) Suggested additions and changes to CCITT Recommendations X3/X28/X29 (Annex B).
(iv) How to integrate X3/X28/X29 with the Transport Service.
(v) How to implement Private PADs.

The Green Book attempts to address several different problems as listed above. Some of these are intended to recommend ways for PTT PADS to develop [(ii) and (iii)], and the Post Office has incorporated the new profiles into the PSS PAD and put forward the suggested changes to CCITT for study in the next period. The remaining recommendations largely affect interworking between hosts (including Private PADs).

2. The Green Book is a Compromise

The need for a recommendation like the Green Book arises from the conflicting requirements of hosts to have terminals coming in from the network access them in the same way as directly connected terminals, but to use their existing terminal support in order to call other hosts. The ideal solution to this problem would be a Virtual Terminal Protocol; however, we have X3/X28/X29 as a major protocol which everyone will implement for PSS.

Therefore the Green Book recommends a "way of working" for host operators to use X3/X28/X29 so that host-host working will be possible. This involves limiting the number of X3 parameters which the host modifies and goes some of the way toward producing a "model" of a network terminal.

Another problem is that X3/X28/X29 does not use a Transport Service. Therefore the Green Book envisages the use of both raw X25 and Transport Service working for some time to come, and also proposes an encoding for X3/X28/X29 messages for the Transport Service.

3. Revision of the Green Book

This version of the Green Book was written and circulated for comment before final versions of the PSS Technical Guide or the Yellow Book (Transport Service) were available. Hence it is now proposed to revise the recommendations to take account of these, and of comments made.

A major problem which the group will have to consider is how far to use non-CCITT facilities (bearing in mind that the PSS implementation represents an attempt to match the next CCITT recommendation rather than the currently published one). The compromise on using both raw X25 and the Transport Service is likely to continue, as is the compromise between ease of use and simpler interworking on the one hand, and the multiplicity of host styles of use and the CCITT Recommendation on the other. In addition future uses must be considered.

Common Questions About the Green Book

Questions:

(i) Why not just use Standard X3/X28/X29?

(ii) Why does the Green Book use the Telenet Parameters?

(iii) Why not do a Virtual Terminal Protocol?

Answers:

(i) There is no such thing. To our general benefit, PSS is an approximation to the next version of X3/X28/X29 rather than the previously published version, which is followed by Euronet and Transpac. Telenet seems to use a mixture of XXX and its original ITI parameters.

(ii) The PSS implementation is based on Telenet and thus their parameters represented what it was possible to do. Apart from answer (i), the major use of the parameters was in the profiles (Annex B) and these are local.

(iii) X3/X29 will be the major interactive protocol on PSS because it is implemented in the PAD. Any VTP we tried to do would have only minority usage and, in any case, I feel that the Euronet DEVT is a perfectly adequate interim "standard" for use until an internationally agreed VTP is available. (Note: Following expressions of interest during Networkshop, it is likely that the DCPU will circulate the Euronet DEVT.)

5. Implementation Issues

This was discussed in greater depth at the last Networkshop, but, in view of the comments on implementation in the first session, it is worth looking at the proposed methods of implementation in a Host (fig. 1), and in a gateway between an X25 network (e.g. PSS) and a local network (which may or may not be X25) as shown in fig. 2.

An object is that conversion is done between raw X29 and TS29, so that in fig. 1 the application interface supports only TS29 and in fig. 2, the whole of the local network uses only TS29. Although I have drawn the figures in the preferred way, the reverse picture with applications using X29sub would be possible.

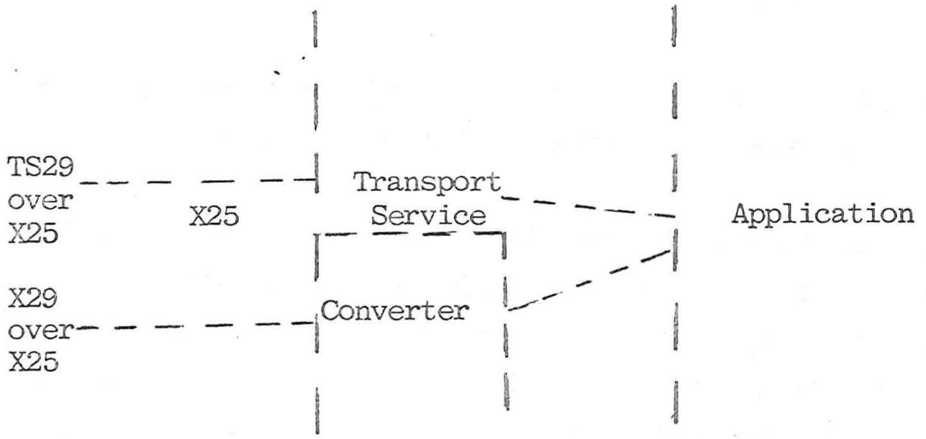


Figure 1: Implementation in a Host

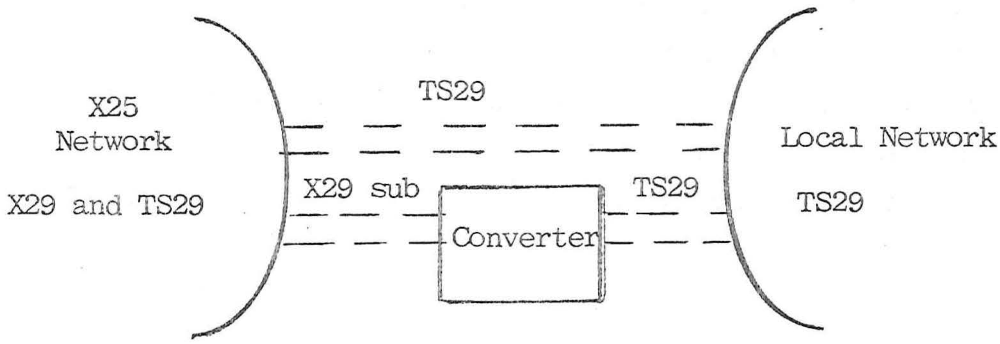


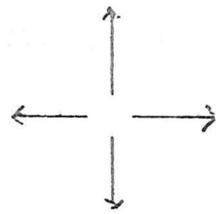
Figure 2: Implementation in a Local Network

5. Future Issues and Screen Editors

There are some classes of use which are not well covered by the Green Book, or any other proposals, and I would like to end this talk by mentioning some of these with a view to getting reactions and possibly study of these in the future.

A major future use in universities is likely to be in "Screen Editors". Some users are already becoming accustomed to these highly interactive editors for local use and word processors. For example, the UCL variant of this uses a relatively cheap terminal (actually a customised Newbury 7009) with special keys for functions such as:

Delete line	Delete word
Join line	Next word
Split line	Back word



It would be unrealistic to use this over PSS because it uses remote echo and the delays and costs would be too high.

There are a number of solutions to this problem; one would be a terminal (or a special system or PAD) which would do the majority of edits locally on the screen, and communicate only updates to and from the mainframe. If it is not possible to use screen editors across networks, then a large number of users may opt to FTP a file, edit it locally and then FTP it back. There is obviously some amount of edits for which the FTP option is cheaper and it is probably around the point at which the whole file is read during the interactive edit. However, I have been unable to draw any detailed comparisons and will merely make the point that this is an area where styles of use could have serious economic consequences (e.g. FTP'ing for one edit).

FILE TRANSFER PROTOCOL STATUS REPORT

Dave Rayner

National Physical Laboratory

FTP Revision

by D Rayner, NPL

1. Introduction

The Network Independent File Transfer Protocol was defined in the "Blue Book" in December 1977. Since then, it has been implemented on a a variety of machines and used over a variety of networks. In particular, five independent implementations have been interworked in various combinations over the UK Experimental Packet Switched Service (EPSS) and attached networks (specifically ARPANET and the NPL network). The experience gained from these and other implementations has been coordinated by the DCPU FTP Implementors' Group. As a result a large number of ambiguities, several deficiencies and a few serious problems have been discovered.

The group's initial response was to produce a list of Recommendations and Reminders [2], updated and reissued about quarterly, and a speculative list of Suggested Extensions [3]. However, with the forthcoming opening of the British Post Office's Packet Switched Service (PSS) it was seen to be desirable to make a maintenance revision of the protocol before it became widely adopted in the UK for use on PSS. The group therefore set up a Working Group on Revision to produce a revised protocol with the aim of minimising the number of incompatibilities introduced whilst producing a protocol that could remain stable for about five years, pending the emergence of an internationally agreed long-term standard.

This paper gives an outline of a document giving the details of all the changes, extensions and clarifications that have been made to the protocol. The document is being produced as information for existing and intending implementors, prior to production of the revised "Blue Book" which will be published later in 1980. It is possible to produce this document on a shorter timescale because it assumes a full working knowledge of the original "Blue Book", simply recording the decisions of the Working Group.

1.1 Types of Attribute

The "Blue Book" describes attributes as all being contained within the Standard Conceptual Filestore at Q. This has never been a very satisfactory picture for the Operator and Monitor Messages or Protocol Identification. It is now recognised that there is a distinction to be made between an attribute applying to the storage of a file at Q and a similar attribute applying to the transfer of the file between P and Q. For instance, P might wish to specify that it is transferring a text file to Q in IA5 but that Q should store it in EBCDIC, thus requiring Q to perform the conversion.

There are, therefore, in practice three types of attribute: those that refer to the storage in the Standard Conceptual Filestore at Q, those that refer to the transfer, and those that provide information and hence refer to either P or Q depending on the command concerned. These will be referred to as Q, Transfer (T) and Informational (I) attributes, respectively.

In the few cases where attributes are needed in both Q and T forms, the distinction has been made by adding new attributes. It is felt that adding new attributes is preferable to adding some new mechanism to indicate the type of attribute in this sense. In these cases, the existing attribute number will refer to the most appropriate form to maintain compatibility, which turns out to be the T form.

1.2 Simple Negotiation

Many of the problems that have arisen with the use of the protocol have been caused by difficulties in using the simple negotiation mechanism with attributes to which it is not suited. Part of the solution has been to fragment and redefine some of the attributes to make their negotiation simpler. For many attributes, however, there is a distinction to be made between values that express what an implementation is capable of and those that express the essential requirements of user or implementor for a particular transfer. These are referred to as capabilities and requirements, respectively.

With the new attribute definitions, P will usually not need to specify both its capabilities and requirements for any particular attribute. If, however, P specifies its capabilities and Q reduces the value below P's requirements or if P specifies its requirements and Q increases the value above P's capabilities, a sophisticated P can terminate this transfer attempt and issue another SFT with revised parameters, thus using a form of iterative negotiation that requires no new mechanism.

A close analysis of the existing negotiation mechanism reveals that the distinction between capabilities and requirements is only important for bitfield attributes and that the use of the operator field in the parameter qualifier is, in most cases, sufficient to make that distinction.

The simple form of negotiation may however still lead to invalidation of some attribute values as a consequence of refinement of others. This applies to any inter-dependent attributes.

1.3 Minimum Required for Open Systems Interworking

Any implementation intended for use in open systems interworking should be able to accept the standard default values of all attributes except Mode of Access, Filename, Username, Username Password, Filename Password, Account and Account Password. This gives a minimum level at which any two implementations should be compatible provided that their capabilities for Mode of Access are compatible. The problem with Mode of Access is that an implementation's capabilities will so vary with its intended style of use that it is impossible to pick a value which it is fair to ask every open implementation to support.

2. Changes

2.1 Termination Phase

A STOPACK command has been added to the Level 0 Termination Phase. It will be used by Q in response to a STOP. P must therefore be prepared to accept this command after sending the STOP. When it arrives P should close the TS connection unless it wishes to reuse the connection by sending another SFT.

Q need not be affected by this change because it can still close the TS connection after receipt of the STOP. If Q issues a STOPACK it may of course simply close the TS connection on receipt of another SFT if it objects to such reuse.

2.2 Commands embedded in data

Data phase commands embedded in the data may now occur at any subrecord boundary, not necessarily at the end of a record. Thus Receivers must be prepared to accept MS, CS, SS and ES commands anywhere, but the change need not affect Senders. This change is made to allow marks to be placed anywhere in the file and to allow files that have no record structure to be sent without needing to impose some arbitrary record structure. The changing of codes within a record is also allowed, but is controlled by the new Data Type attribute (see below).

2.3 Text Formatting

The old Format Effectors attribute has been replaced by the following bitfield attribute, in which the compatibility of the default and values [0001] and [0002] is maintained.

Text Formatting default value [0001]

[0001] End of record implies new line action. No actions are implied by embedded characters.

[0002] Each record starts with an ANSI control character.

[0004] End of record implies new line action. The formatting actions CR, LF, NL, BS, FF may be represented by embedded characters.

[0008] End of record implies new page action. The formatting action NL may be represented by embedded characters.

[0010] End of record implies new page action. The formatting actions CR, LF, NL, BS, FF may be represented by embedded characters.

[0020] No formatting action is implied by end of record, but the formatting action NL may be represented by embedded characters.

[0040] No formatting action is implied by end of record, but the formatting actions CR, LF, NL, BS, FF may be represented by embedded characters.

[0080] No formatting actions are implied by end of record or by embedded control characters.

The negotiation should reduce the value of this attribute to exactly one bit being set, otherwise there is a protocol error.

The formatting actions will be fully defined in terms of their effects upon a a matrix of characters.

Note that Horizontal Tabs are now covered by a separate, orthogonal, attribute (see below) and that Vertical Tab has been removed as a defined formatting action.

2.4 Text Codes and parity

2.4.1 Text Code for Transfer attribute

The Codes attribute has been replaced by the following Text Code attribute.

Text Code for Transfer   default value [0001]

[0001] IA5
[0004] EBCDIC
[0008] Private code - see Private Code Name attribute.

All other bits are reserved.

Mixing of more than one text code in a file is not allowed (see the new Data Type attribute, 2.5.1 below). However, the final value of this attribute can specify more than one code, because the CS will resolve any ambiguity.

It is no longer possible to negotiate about parity bit options; the appropriate bits have been omitted to simplify the negotiation of this attribute. Implementors are recommended to ignore parity when using IA5, but Senders will still be able to specify on the CS the way in which the parity bit will be set.

Note that the default for this attribute means IA5, parity ignored, which is the same meaning as the old default for the Codes attribute.

2.4.2 Code Select command

The parity bit setting, specified in bits 5 to 8, has been extended to include "always one", as follows:-

[00] Any parity (i.e. undefined and ignored)
[20] Always one
[40] Odd parity
[80] Even parity
[C0] Always zero

These bits should be [00] if the code specified does not use parity. So they should always be [00] for EBCDIC and binary. The Sender should only use a value other than [00] if the Receiver can rely on its accuracy. However, the Receiver may ignore the information if it so wishes.

2.5 Binary Transfers

There are several changes which affect binary and mixed (text and binary) transfers because this is the main serious problem area in the "Blue Book". However, the changes will not affect the large number of implementations that only handle text files. Furthermore, 8 bit binary is affected only by the Data Type attribute.

2.5.1 Data Type

Firstly, a Data Type attribute has been added to replace the binary bit in the old Codes attribute. This new attribute enables one to distinguish between the capability of handling either text or binary files from the capability of handling mixed (text and binary) files. Mixing more than one text code in a file is not allowed.

Data Type default value [0001]

[0001] Text
[0002] Binary
[0004] Text and binary mixed
[0008] Mixed within a record

[0008] should only be used if [0004] is set; if both are set then code changes are allowed on any subrecord boundaries. Otherwise code changes can only occur on record boundaries. An example of where mixed text and binary might be needed within a record is a graphics file that might use records to represent lines or pages.

2.5.2 Binary Format

The Binary Mapping attribute is split into Binary Format and Binary Word Size, as follows.

Binary Format default value [8002]

[0001] Words packed with no gaps
[0002] Words aligned on octet boundaries

[4000] High-order octet first
[8000] Low-order octet first

This attribute must be reduced to exactly 1 bit in each pair. The default is aligned, low-order first.

2.5.3 Binary Word Size

Binary Word Size   Integer, default 8

Implementors are warned that agreement to a size other than 8 will involve the use of a different subrecord analysis, as described below.

2.5.4 Binary Subrecords

Binary subrecords will use counts of binary words rather than octets, the length in octets being the minimum number of octets to contain the given number of words in the format being used. Therefore, for 8 bit words and for aligned words of less than 8 bits sent low-order first the subrecord structure is compatible.

2.5.5 Binary Compression Facility

Finally, the Facilities attribute has been extended to include "binary compression", [0040]. The "data compression" bit, [0001], then becomes "text compression". Implementors should be warned that the compression facility bits are only meaningful for appropriate Data Types (e.g. if the Data Type is text then binary compression can be ignored).

2.6 Protocol Identification

Protocol Identification will change to reflect the new version of the protocol. The revised protocol, FTP-B(80), has been assigned the value [01] in the first octet. The allocation of the second octet is a local matter for implementor use as before. The default is "no value available", but FTP-B(80) should be assumed.

2.7 Infinity

Hex FFFF is now defined to represent infinity for numeric attributes.

2.8 Maximum Transfer Size Default

The Maximum Transfer Size default has been changed to [FFFF] (infinity). This attribute is to be used as a long-stop. An attempt to transfer more than this limit is considered to be a protocol error. However, if the default is used, no upper limit should be applied. A new attribute for Estimated Transfer Size (see below) has been added to remove the previous ambiguity in the meaning of Maximum Transfer Size.

2.9 Maximum Record Size for Transfer Default.

The default for the Maximum Record Size for Transfer attribute will be [FFFF] (infinity). When working with this default, implementors should not expect to buffer complete records.

2.10 Acknowledgement Window Default

The default for Acknowledgement Window will be changed to 255 (the maximum it can take) to avoid Receivers having to acknowledge marks before they have been able to store the data securely. Problems can arise with small window sizes if the Sender sends marks so close together that the window becomes full before the Receiver has filled one storage (e.g. disc) buffer. This change does not overcome these problems but attempts to mitigate them.

2.11 Code Values for State of Transfer Attribute

The codes for the State of Transfer attribute have been revised as follows:-

VIABLE

[0000] Acceptable and satisfactory

REJECTED

[1001] See text message
[1002] Unacceptable transfer control attribute settings
[1003] Acceptable but deferred
[1004] Resumption impossible

TERMINATED

[2000] Satisfactory
[2001] Problem - see text message

ABORTED

[3010] No retry possible
[3011] Retry possible

2.12 Messages

Operator and Monitor Messages have been renamed Action and Information Messages respectively. These parameters can in future be sent more than one to the command so that messages longer than 255 characters can be sent.

2.13 Kinship

The Kinship attribute has been removed. It will not be replaced.

2.14 Special Options

The Special Options attribute will be allowed to take either string or numeric values.

2.15 Parameters on GO

The Facility for "parameters on GO" has been removed and that bit, [0020], has become reserved.

3. Extensions

3.1 Resumption of an old transfer

The "Blue Book" contains a Facility "later resumption of this transfer in new transfer possible". Unfortunately, it was not clear how a resumption of an old transfer could be distinguished from a brand new transfer. An extension has, therefore, been made to enable transfers to be resumed as often as necessary without any ambiguity.

Mode of Access has been extended to include the assignment of bit 8, [0100], to "resumption of an old transfer". This is orthogonal to all the other bits in Mode of Access. When it is used, the Transfer Identifier should be the same as it was for the original transfer.

It will not usually be necessary to resume from the "end of file", since errors occurring after the data transfer phase will not usually be recoverable by resumption. If, however, such a resumption is desired then the Initial Restart Mark should be [FFFF] to mean "end of file".

3.2 New Attributes

The following new attributes have been added as extensions:-

1. Text Code for Storage
2. Private Code Names
3. Page Width
4. Page Length
5. Horizontal Tabs
6. Device Type Qualifier
7. Estimated Transfer Size
8. Record Preservation
9. Maximum Record Size for Storage

3.3 Output Device Type

The list of values has been extended as follows:-

"LP" Line printer
"CP" Card punch
"PL" Graph plotter
"TP" Paper tape punch
"COM" Micro-fiche

References

[1] A Network Independent File Transfer Protocol, prepared by the UK High Level Protocols Group, HLP/CP(78)1, 12 December 1977, available from Dr P.F. Linington, Data Communication Protocols Unit, Computer Laboratory, Corn Exchange Street, Cambridge CB2 3QG.

[2] Recommendations and Reminders to Implementors, produced by the Data Communication Protocols Unit FTP Implementors' Group, edited by D. Rayner, 19 February 1980, available from Dr D. Rayner, DNACS, National Physical Laboratory, Teddington, Middx.

[3] Suggested Extensions to FTP, produced by the Data Communication Protocols Unit FTP Implementors' Group, edited by D. Rayner, 4 October 1979, available from Dr D. Rayner, address as in [2].

HIGH LEVEL LANGUAGES AND COMPILERS FOR NETWORK IMPLEMENTATION

John Davies

University of London Computer Centre

HIGH-LEVEL LANGUAGES AND COMPILERS FOR NETWORK SYSTEM IMPLEMENTATION

John I. Davies
Network & Communications Group
University of London Computer Centre.

A paper presented at Networkshop 6.
University of Durham April 1980.

38

INTRODUCTION

Each generation of network software projects produces a renewed series of declarations that software should be portable and shareable. These declarations have resulted in very little action. One of the major reasons for this has been the difficulty involved in using externally developed software as a result of relatively poor documentation and the lack of tools with which to manipulate imported software.

The most striking deficiency from the network programmers tool-kit has been that of efficient high level language compilers. This deficiency affects not only the network software itself but also the production of further software tools. This multiplying effect can be observed in the use of mainframe macro-assemblers to generate system configurations to be mounted on minicomputers. The programmer must thus be familiar not only with the assembly language of his target machine but also with the macro-language of his mainframe.

This presentation is based upon a study of available high-level languages and compilers conducted for the Network and Communications Group at ULCC. The group is currently engaged in a number of software projects using both assembler and high level languages.

ADVANTAGES

The most significant advantages of high-level language programming for systems software are in the area loosely describable as "software management". A source language program has an intelligibility roughly proportional to the regularity of the language in which it is expressed. This language reflects the structure of a real (assembly language) or imaginary (high-level language) processor.

High-level languages are more regular than assembly languages primarily because real machines are so irregular in their structure. (This accounts for the comparatively intelligible nature of PDPll assembler).

A source language program produces an executable program whose security is roughly proportional to the degree of cross-checkable redundancy in the source language and in the object language. Object language cross-checking must always be paid for in hardware or execution-time software overheads. Source language cross-checking need only occur during compilation and may cost very little. High-level languages provide substantial degrees of cross-checkable redundancy whilst assembly languages provide virtually none.

Thus the primary advantages of high-level languages programming are intelligibility and security. Software written in a high-level language is thus more quickly produced, more robust in operation, more easily maintained, and more easily modified than equivalent software written in assembly language.

Two additional issues often intrude into the high-level language versus assembler debate. First, high-level language programs are claimed to provide portability across a number of machines. This is only partially true even for application programs. System programs may be portable where they service functions which are distant from the machine or operating-system dependent interface but as the interface is approached, software becomes more and more machine dependant, irrespective of the language in which it is written.

Second, assembly language is said to provide opportunities for highly-optimised use of the real-machine order code whilst the code produced by high-level language compilers is necessarily less efficient. This may almost always be demonstrated by taking selected sections of code produced by a high-level language compiler and tuning them further. However, experience shows that assembly language programmers produce code of about the same bulk and speed as that produced by a high-level language compiler with fairly simple optimisation capabilities. (This assertion is based upon experience with the IMP compilers at ERCC Edinburgh).

Why have system programmers, and network software implementors in particular, been so relucant to take up high-level languages?

The technically legitimate reasons can be summarised as efficiency and access to hardware. The first of these should no longer be a problem given modern optimising compilers. The second is very easily remedied in a crude manner by providing the facility to embed assembly language statements. A number of more sophisticated techniques have also been implemented.

One of the peculiarities of network software is the complexity of the decisions to be taken in handling protocols. This is, in many ways, paralleled in the syntax analysis phase of compilers. The result, in both cases, has been a tendency to use a table-driven strategy. This inevitably produces some move away from the direct use of high-level languages. However compiler-writers have not discarded high-level languages but have built table generators as an additional software tool. Some network system implementors have taken the same course but many more have allowed table-driving to become another excuse for implementing yet more obscure software.

LANGUAGES

What do we require of a high-level language in general, what are the special requirements of system software, and how do the requirements of network software differ from those of other varieties of system software? The most widely accepted set of high-level language features are those associated with Algol 60 and its descendants. These are nested scoping of names, nested control flow structures, and clear typing rules. A number of additions have been made to the basic set provided by Algol 60. These have involved extensions to the scoping rules by the provision of simple over-rides such as access to separately compiled entities or the more complex facilities mostly deriving from the Class feature of Simula 67 and present in languages such as Concurrent Pascal, Modula, Modula II, Pascal Plus and others. Extensions to control flow have introduced a clearer nesting for multiway branches, replacing the crude switch vector of Algol 60, and allowed some relaxation of nesting without the necessity for arbitrary jumps by the provision of such things as return statements in procedures.

This quick summary outlines my basis for the evaluation of the various languages by their visible features. System programming demands a number of features in addition to those provided for applications work but the demands here are much less widely agreed.

I have summarised the visible features of six system languages plus Pascal in the attached table (Fig. 1). I shall not attempt to evaluate these features any further. Their evaluation may be a useful activity if conducted in a systematic manner but the differences between the languages presented here are, from the practical network implementors point of view, less important than matters relating to compilers.

COMPILERS

Machines not being programmable directly in a high-level language, compilers are a necessary evil. The sort of compilers required by network implementors differ considerably from those required by applications programmers. Questions of efficiency and access to hardware, as already discussed, are often important but these appear in different forms according to the environment in which the network software resides.

I am mainly concerned with the programming of dedicated mini and micro computers to perform network functions. Network software residing in mainframes is susceptible to the same sorts of arguments in favour of high-level language implementation but the constraints are very different from those in dedicated systems. The majority of dedicated-system network software is developed on mainframes using cross-compilers and cross-assemblers. These software preparation tools are often themselves written in high-level languages. This raises the possibility of making such tools portable. A list of known compilers and the source languages in which they are written is included here (Fig. 2) together with a list of systems known to support the source languages in which compilers have been written (Fig. 3).

DECISIONS

We have made our decision on implementation language at ULCC. We shall use BCPL for all major projects for the forseeable future. The most important single reason for this has been the desire to use the same language on both the LSI II and the Modcomp 7860. This restricted the choice to BCPL since this appears to be the only mature system language available for the Modcomp. The major disadvantage of BCPL, the absence of any typing constraints, is considered worth bearing in the interests of a common language.

We are continuing to watch developments in advanced system languages and, if manpower permits, will implement one of our less urgent LSI 11 subsystems using Modula II as a pilot project. In the long term ADA seems to be unstoppable although it is likely to demand a not insubstantial run-time system (to be written in BCPL or Modula II?).

CONCLUSION

There are sufficent mature compilers for high-level system languages to allow some choice of language to the network system implementer. Efficient code generation and good access to hardware make the excuses for continued assembler programming look thin. The majority of network machines are small and dedicated to a specific task and compilation usually takes place on another, larger machine. The compiler, assembler, and linker/system-generator are all examples of software tools and ability to move these tools from mainframe to mainframe is a vital requirement for the sharing of software and experience within the networking community. As a first step towards this a catalogue of such tools (all of which should be written in high-level languages) needs to be compiled. Early consideration should be given to limiting the number of languages in which such tools are written in the interests of tool portability and wide intelligibility. I shall not make any suggestions as to which languages might be included. The attached tables (Figs. 1, 2 and 3) describing the available languages and their features are clearly incomplete and I would welcome both additions and corrections.

Appendix	-	Language Definitions or Descriptions
ADA	-	Sigplan Notices Vol. 14 No. 6 June 1979 (2 parts)
BCPL	-	"A Proposed Definition of the language, BCPL" M.D. Middleton Das RZ der Universitaet Regensburg Universitaetsstrasse 31 D-8400 Regensburg W. Germany
C	-	"The C Programming Language" D.M. Ritchie, B.W. Kernighan, and M.C. Lesk Bell laboratories, Computing Science Technical Report Number 31 October 1975
Concurrent Pascal	-	"The Architecture of Concurrent Programs" Per Brinch Hansen Prentice-Hall, 1977
Coral 66	-	"Official definition of Coral 66" P.M. Woodward et al H.M.S.O., uncertain date
IMP 77	-	"The IMP 77 Language" Peter S Robertson University of Edinburgh Department of Computer Science Internal Report CSR-19-77, 1979
IMP (EMAS)	-	"The IMP Language and Compiler" P.D. Stephens Computer Journal Vol. 17 No. 3 1974
Modula (Original)	-	"Modula - a language for Modular Multiprogramming" N. Wirth Software Practice and Experience Vol. 7 No. 1 1977

Modula (York)	-	"Functional Specification of the Modula Compiler" I.D. Cottam University of York Department of Computer Science Heslington, York 1979
Modula II	-	"Modula II" W. Wirth. H. Oswald and H. Seiler ETH Zurich, January 1980
Pascal	-	"Pascal User Manual and Report" K. Jensen and N. Wirth Springer - Verlag 1975
Pascal Plus	-	"Pascal Plus-Another Language for Modular Multiprogramming" J. Welsh and D.W. Bustard Software Practice and Experience Vol. 9 Page 947 1979
RTL2	-	"RTL2 Languages Specification" Imperial Chemical Industries Ltd. 1974 (available from SPL, 12 Windmill St., London W1)
Simula 67	-	"Simula BEGIN" H. Birtwistle et al published by Auerbach 1973

LANGUAGE	BLOCK STRUCTURE	SERIAL FLOW CONTROL	LOOP FLOW CONTROL	SCALAR TYPES	TYPING OVERRIDES	COMPLEX TYPES	INDIRECT
/ RTL2	TWO LEVEL EXTERNAL	ITE SWITCH	INCREMENT WHILE	INTEGER BYTE	EXPLICIT SHIFTMASK	ARRAY RECORD	REF
/ CORAL66	MULTILEV	ITE SWITCH	INCREMENT WHILE	INTEGER	EXPLICIT SHIFTMASK	ARRAY 'TABLE'	POINTER
/ BCPL	MULTILEV EXTERNAL	ITE/UTE CASE +DEFAULT	INCREMENT WHILE/UNT EXIT/CONT	MC WORD MC CHAR	UNCONSTR- -AINED	VECTOR	POINTER MC ADDR
/ IMP77	MULTILEV EXTERNAL	ITE/UTE SWITCH	INCREMENT WHILE/UNT EXIT/CONT	INTEGER BYTE	EXPLICIT SHIFTMASK	ARRAY RECORD	REF MC ADDR
/ PASCAL	MULTILEV	ITE CASE	INCREMENT WHILE/UNT	INTEGER BYTE BOOLEAN SUBRANGE	NONE	ARRAY RECORD VARIANT	REF (HEAP)
/ MODULA (YORK)	MULTILEV	ITE CASE +DEFAULT	INCREMENT WHILE/UNT EXIT	INTEGER BYTE BOOLEAN	PERIPH REGISTERS	ARRAY RECORD	NONE
/ MODULAII	MULTILEV EXTERNAL	ITE CASE +DEFAULT	INCREMENT WHILE/UNT EXIT	INTEGER BYTE BOOLEAN SUBRANGE	EXPLICIT	ARRAY RECORD VARIANT	REF (HEAP)
/							

MAINFRAME SYSTEM				
CDC NOS	PASCAL	BCPL		
CDC SCOPE	PASCAL	BCPL		
IBM OS	IMP	PASCAL	BCPL	RTL2
ICL VME/B	IMP	PASCAL	RTL2	
EMAS 2900	IMP			
MULTICS	C	PASCAL		
UNIX11	C	PASCAL		
RSX11	PASCAL			
TOPS10	PASCAL	BCPL	IMP	
VAXUNIX	C			

FIGURE TWO
KNOWN
MAINFRAME
SUPPORTED
LANGUAGES

TARGET MACHINES	RTL2	CORAL66	BCPL	IMP	MODULA	MODULAII
PDP11	RTL2	-	BCPL	IMP	BCPL	PASCAL
PE 32BIT	RTL2	-	-	IMP	-	NONE
MODCOMP	NONE	-	BCPL	NONE	-	NONE
GEC4000	RTL2	-	-	-	-	NONE

FIGURE THREE
KNOWN
COMPILER
IMPLEMENTATIONS

/

USER'S VIEW OF ACCESSING SERVICES OVER NETWORKS

Adrian Stokes

Hatfield Polytechnic

A User's View of Accessing Services Over Networks

by

Adrian V.Stokes

The Hatfield Polytechnic

1. INTRODUCTION

In the past, users of networks have been, in general, computer science specialists; usually, in fact, people doing research into networks themselves.

With the significant increase in access to networks over the last few years and the potentially vast number of users of services over networks, it is clear that problems will arise. These problems are not different from those encountered in using a single computer system but simply become more apparent in a network context.

In this paper, we consider the problems involved with use of a computer at a single site and then examine related problems in a network context. We then propose some (perhaps ideal) solutions.

2. PROBLEMS AT A SINGLE SITE

At the Hatfield Polytechnic, we have an interesting user population which probably does not occur at many other sites and we have performed various surveys [1] to examine their mode of usage of the computer and the problems they encounter.

The environment we are discussing is that there are a large number of students and staff (of the order of 5000) who nearly all use the central computer (a DECsystem-10 under the TOPS-10 operating system) but who are, in general, non-computer science specialists. There are close analogies that can be drawn between this user population and that of network users in the next few years.

In general, it was found that the majority spent little time in executing their own programs but spent much time either executing packages (either provided by the system or by a teaching member of staff) or using system facilities such as an editor.

They encountered many problems and the major causes of these problems can be specified as follows:

* The interfaces to the various parts of the system are non-uniform

* The "HELP" facilities are, in general, not very helpful

* The current environment of the user is not always obvious

* The syntax of the command language is inconsistent

To give a few simple examples of the problems, let us examine some errors actually discovered during the surveys:

(i) Issuing a command 'DIRETORY' to the Monitor (containing a simple typographical error), gave the result:

?DIRETORY?

Even worse, but derived from the same source, a naive user wishing to obtain help facilities tried typing a single question mark and was rewarded by:

???

(ii) The syntax of commands in BASIC (the most common system used by the user population we are considering) gives rise to such problems as the following, which require little comment:

```
READY                          BASIC prompt
REPLASE                        Simple spelling mistake
?FILE LASE NOT FOUND
READY
```

and

```
READY
CATOLOG
?NO SUCH DEVICE OLOG
READY
```

(iii) In order to escape from such an unfriendly system, a naive user might think that the opposite of 'LOGIN' was 'LOGOUT' but, in fact, the simplest way of exiting from the system is to type 'K' (KILL). Many users seem unable to remember this and the survey showed a number who went back into BASIC and typed the BASIC command 'BYE' which performed the required function but at significant system overhead costs.

3. PROBLEMS IN A NETWORK ENVIRONMENT

As we said above, the problems associated with access to services in a network environment are little different than those for a single system but may be an order of magnitude more serious.

For example, using ARPANET, in order to connect to the London PDP-9 via a TIP, it is necessary to type:

@O 42

(that is, "Open a connection to Host 42"). A worse example was given by the method required to connect to the National Library

of Medicine, required during an experiment funded by British Library for library staff to access databases on that machine [2]. The sequence of commands required was:

```
@H 147
@R F S 14400003
@S T S 14400002
@F
@E L
@P B
```

Admittedly the connection was experimental and such a sequence would not have been required in a production environment. Needless to say, the problems encountered by that user population were significant, especially as the error messages returned from the above commands were almost totally meaningless.

A final, brief example concerns the addressing structure of networks. In the case of EPSS, the network address for the interactive system on the DECsystem-10 at Hatfield is 13250300, once again, not easily remembered.

Not only is access to Hosts on networks difficult due to the user interfaces supplied but other problems arise. The most obvious is that there is a wide disparity of operating systems on the various computers attached to a network and the user, who is bewildered enough on a single computer system, has even less chance of being able to cope, particularly in error situations. This is exacerbated by the lack of suitable documentation; it is clearly not possible for any user site to have available all the documentation needed for a network. Indeed, there are considerable difficulties in even keeping up-to-date with the machines available on the network, let alone finding out about, perhaps minor, changes to the systems on those machines.

4. SOME PROPOSED SOLUTIONS

With such a plethora of problems, it seems that users are consigned never to venture away from their local (friendly?) computer. But there are some solutions which could be applied and, in this section, we consider some of these solutions, together with the additional problems that they bring. The first solution is to make the interface to the network from each user system "user-friendly", for example, by being able to specify Host names rather than some complex number. This raises an additional problem, namely that there is often no commonly used name for a site and so one Host may be known by different names from different user sites. This problem can be overcome easily by having a centralised information source ("Network Information Centre") whose function is to maintain accurate records for the network and which is consulted regularly by system implementors.

Secondly, it is vital to have adequate, context-dependent help facilities at the user site so that, for example:

?	gives a list of available commands
CONN ?	gives a list of Hosts

and, in addition, error messages must be intelligible without being verbose.

Having accessed the remote Host, the user has similar problems and the first solution to those is, once again, adequate, detailed, context-dependent help facilities. Secondly, there is a need for a single, unified interface to every Host system. Considerable work has been done in this area (standardisation of control languages and so on) but it appears impracticable for the foreseeable future and so a compromise solution is to provide a system on the user machine to give the required interface, a "Network Access Machine" [3]. These have been built already but, due to the complexity of Host operating systems, can only handle relatively simple interactions. On the other hand, one should not lose sight of the fact that many users of networks will only require simple facilities.

5. SUMMARY

Most users of computer network facilities are going to be "naive users" and not computer science specialists. Therefore, the interface to the network must be as simple as possible to use with context-dependent help facilities, clear error messages and it must be consistent for all Hosts. The technology exists for this to be done; in order for networks to be used fully, it is vital that it is done.

REFERENCES

1. Huckle, B.A., "Results of a Survey on User Behaviour on the DECsystem-10", Report TM-1, The Hatfield Polytechnic, June 1977.

2. Stokes, A.V., "Access to an Information Retrieval System via a Distributed Computer Network", INDRA TR-40, Department of Statistics and Computer Science, University College London, August 1977.

3. Rosenthal, R.M., "Network Access Techniques - a Review", Proc. AFIPS NCC, 45, 495-500, 1976.

POST OFFICE DEVELOPMENT AID

Ian Service

University of York

Discussion on the Post Office Development Aid

This session was originally intended to be given by users with experience of using the development aid, but there were none. The announcement by the Post Office that the Development Aid would not be available until September ensured that this session was thin on content, especially as the announcement was only made hours before the session.

However it seemed worthwhile continuing the discussion and extending it into related topics. Some of the main points of the session were:-

- a description of the development aid by Pat Morrison (P.O.) and some discussion on how users would like to use it. It turned out that many people would not be inconvenienced by the development aid not being available until September.

- a discussion on development aids from other sources. For example, the Tekelec being bought for the community and being delivered to ULCC.

- general discussion on debugging aids and requirements for testing.

On the whole this discussion was fruitful and interesting. The allotted time was soon used up - an indication of how important development aids are?

J D Service
University of York

NETWORK OPERATION AND MANAGEMENT IN WIDE-AREA NETWORKS

Paul Kummer

Daresbury Laboratory

MANAGEMENT AND OPERATION OF A WIDE-AREA NETWORK

P S Kummer

Network Development Group
Science Research Council, Daresbury Laboratory

1. INTRODUCTION

The SRC network has developed over the past few years from its original star configurations centred on the two Laboratories to a UK-wide packet-switched system. This evolution was required by the SRC's decision to allocate time on the main computer systems by discipline rather than geographic location.

This paper will discuss the functional requirements for a network monitor system capable of providing the necessary tools to operate and manage the SRC network. Many of the requirements apply equally well to local area networks, and many will seem obvious. However, it is very easy to overlook the obvious and it is my intention to detail all of the requirements so as to indicate the effort necessary to implement an adequate network monitor system. My own feeling is that this effort is at least equal to the effort required to implement the network in the first place.

The management aspects of the SRC network will not be detailed here as they are presently being discussed. However, good management will require adequate tools within the network and these will be discussed with respect to functions to be provided in the network monitor centre.

2. OPERATIONAL REQUIREMENT FOR A NETWORK MONITOR CENTRE

There are several points which require the installation of a monitor centre on the network:

(a) The rapid growth of the network

(b) The increasing sophistication of user demand.
- we have developed the network to provide "universal access" so we should not be surprised if that is how it is used.

(c) The proliferation of computer installations.
- the SRC is installing "large minis" in university departments (to act as enhanced workstations with local editing, batch and graphics facilities) which are hosts in their own right.

(d) The network is now being used by many different disciplines (with different funding bodies) and there is a requirement for more active accounting of network use.

3. CHARACTERISTICS OF WIDE AREA NETWORKS

There are several points which are characteristic of wide area networks:

(a) The network is geographically disperse - there are components off-site.

(b) There are switching exchanges off-site - possibly running without supervision.

(c) There are users off-site and often many components between a user and a service.

(d) There is a reliance on others (e.g. the Post Office) for communication lines. These lines are error prone.

(e) The communication lines are slow. In particular, the error recovery procedures are slow and will result in poor response on "noisy" lines.

(f) There are many cross-connections at level 3. One of the objectives of a packet switching network is to reduce the cost of communication lines which leads to a level 2 topology as shown in figure 1a. However, at level 3 we will see something more like figure 1b.

4. NETWORK COMPONENTS

For the purpose of the following sections we can define the network to be built from the following components:

Switches		
Hosts		
Gateways		
Links		
Access facilities	- e.g.	PAD
		RJE stations
Logical connections	- e.g.	level 3 calls
		transport service
		high level protocol used
Services	- e.g.	time-sharing
		RJE
		File transfer
Ancillary devices	- e.g.	terminal
		line printer
		spool file

5. NETWORK OPERATION FUNCTIONS

AIM - To provide a set of functions to maintain, automatically, or with the help of operator intervention, a high quality of service to the users.

The following functions can be identified. They will be explained in greater detail in later sections.

1. Automatic failure detection
2. Automatic failure recovery
3. Failure prediction
4. Load monitoring
5. Statistics recording
6. Operator commands
7. Online status information
8. Network-wide broadcast.

6. SOME CONSTRAINTS

6.1 Centralised vs. Distributed system.

The network must not depend on the operational status of one single component.

e.g. Authorisation tables should be stored in the individual gateways.

However, a central facility may exist with the capability to change these tables.

Obviously some balance is required between what is done centrally and what can be distributed.

6.2

Operational decisions may be made on external factors (e.g. Money!) so the system cannot be fully automatic.

e.g. It is necessary to notify the Post Office when a communication line fails.

This means that the network will have operations staff, and important consideration needs to be given to the ease and efficiency with which operators may communicate with the system. This gives a requirement for easily interpretable displays.

6.3	We need to let users know what is happening (e.g. when components fail) with minimal operator intervention.

When something goes wrong the operators are usually busy fixing it and have little time to manually answer user enquiries. Thus, we should provide a network status display available from any terminal on the network.

6.4	The efficiency of the network relies on operators knowing what is happening.

This requires that the monitor system be extremely reliable and that the operators must always be able to access it.

6.5	Users must not be able to affect the operation of the network by unauthorised access to the monitor system.

This implies some form of access control, e.g. passwords.

7. FUNCTIONS OF THE MONITOR SYSTEM

7.1 Automatic Failure Detection

The monitor should know the status of every component in the network. There are two possible ways to do this.

(a) Interpret information from adjacent components;

(b) Maintain a "handshake" with the component.

An additional constraint here is:

The failure detection time must be less than the time taken for a user to phone the operators.

One of the situations that may be encountered is illustrated in figure 2 and means that there are three basic states for a component:

ACTIVE
BROKEN
UNKNOWN

7.2 Automatic Failure Recovery

This is not yet seriously planned for the SRC network.

One possibility is the changing of routing tables to bypass a failed component.

7.3 Failure Prediction

This can be done on communication lines by trend analysis of line error rates and may be used to alert the appropriate personnel and attempt recovery before the line actually fails.

Much work can be done in this area.

7.4 Load Monitoring

The following parameters may be measured:

LINES	bytes/packets	read/written
	errors	
CALLS	bytes/packets	read/written
	errors (e.g. RESET, REJECT)	
	duration	
	special facilities used	
	high level protocol used	

SWITCHES	throughput	)
	queue lengths	) average and peak
	processor utilisation	)
	number of active calls	)

These will be used to:

(a) Generate network performance figures

(b) Measure the effect of a change

(c) Predict the future performance of the network

(a) and (b) are required in real-time for operations staff and to aid in fault diagnosis.

(a), (b) and (c) are required for offline analysis to aid long term planning and for accounting.

7.5 Statistics Recording

Data gathered from the network should be recorded for later analysis to produce reports, aid long-term planning and to assist in the investigation of faults reported well after the event.

There is probably a requirement here to keep both a day file and a long term database.

7.6 Operator Commands

This topic requires further study but possibilities include:

Display/change memory
Force dump/reload
Change routing tables
Change authorisation tables
Start/Stop (drain) link
Copy new system through network
Copy dump to central site

It will be essential to keep a record of operator commands entered and their responses in some form of history file.

7.7 Online Status Information

In a large, complex network there is a tendency for users to become isolated. This effect is enhanced when something goes wrong. Unfortunately we cannot always rely on operators to take action to inform users of the nature of a fault.

The monitor system knows the status of the network and thus can generate a text file which may be interrogated from any terminal on the network.

For the main components of the network an "English" message should be generated to indicate the status of that component. This message should be changed automatically on a change of status of the component.

It should also be possible for the operators to enter/edit further messages against each component. This further information may be used for scheduling, and indicating estimated recovery time.

As the SRC network is getting too large for simple "list" access then some form of interactive service is required here.

7.8 Network-wide Broadcast

There are two types of network "broadcast":

PASSIVE - message generated when a user attempts to access a service.

e.g. call connected
failed - host machine
failed - network error

ACTIVE - an operator command to tell users of an imminent failure.

The active broadcast is presently the subject of much debate, the two main points being:

(a) Should the user/service be able to inhibit broadcast messages e.g. during formatted text output.

(b) Should it be possible to only send messages to users going to be affected. This will be difficult !

8. SOME PRACTICAL CONSTRAINTS

8.1 **The gathering of information must not significantly load the network**

Even so, an enormous amount of data will be generated and will need to be reduced to something sensible and meaningful.

8.2 Who needs to know - and when ?

Operators	Seconds
Operations management	Minutes
Maintenance personnel	Minutes/hours
Contractors (e.g. P.O)	Hours
Development groups	Weeks/months
Planning committees	Weeks/months
Funding bodies	Months

9. REQUIREMENTS OF SWITCHES

From what has been said above the following requirements can be identified for the network switches (which will be the main providers of information to the monitor system).

9.1 Maintain a connection to the monitor system.

This connection may use fast select, PVC, or SVC.

9.2 Use a standard protocol over this connection.

e.g. Transport service

or X29 (will allow access from any network terminal as well as from the monitor system).

9.3 Provide the following information.

LINKS
- become active
- statistics (e.g. every hour) (see section 7.4)
- broken, with statistics.

Note - the monitor may receive data from both ends of a link and must be careful of double accounting.

CALLS
- attempt fail/success
  addressing
  facilities
  high level protocol
- statistics (e.g. every hour) (see section 7.4)
- closed/broken, with statistics

Note - as for LINKS, the monitor may receive data on a CALL from more than one service.

INTERNAL LOADINGS
- processor utilisation) average/peak
  queue lengths)

  buffer utilisation, present/high water mark/ total available.

9.4 When the connection to the monitor is established (or on monitor request) provide the following information:

LINK - active/broken

CALL - which LINKS used
logical channel numbers

9.5 Accept operator commands from the monitor (see section 7.6).

10. IMPLEMENTATION

The present implementation plans are based on a PDP11/34 at the Daresbury Laboratory. This system has 192 Kbytes of memory, 20 Mbytes of disc storage, runs under the RSX11-M operating system and our current thoughts are to use MACRO11 and FORTRAN as the implementation languages.

The operator interface will use a RAMTEK 6000 series colour monitor. This follows on from previous work on colour monitors to display the memory and disc utilisation on the Daresbury IBM system. This work has shown the benefit of using colour to display complex status information.

Figure 3 shows the logical configuration of the monitor system. All connections to the monitor system are through the network although, for reliability reasons, some of the connections may require dedicated communication lines.

The present work has three objectives:

(a) To define the network within the monitor system

(b) To identify the information to be kept within the system

(c) To define the necessary processing to provide the required facilities.

This will lead (in about 3 months time) to a firm proposal in terms of:

processor power
disk capacity
cost.

Author's Note

From discussions after presenting this paper I realise that I did not adequately stress the importance of producing a standard for gathering operational information from the network. This standard is required to enable the sensible operation and management of the network which may be built from many types of switches, etc. (In particular, components not built inhouse). The standard needs to define:

(a) The method of connection

(b) The protocol to be used

(c) The format and content of the information transferred.

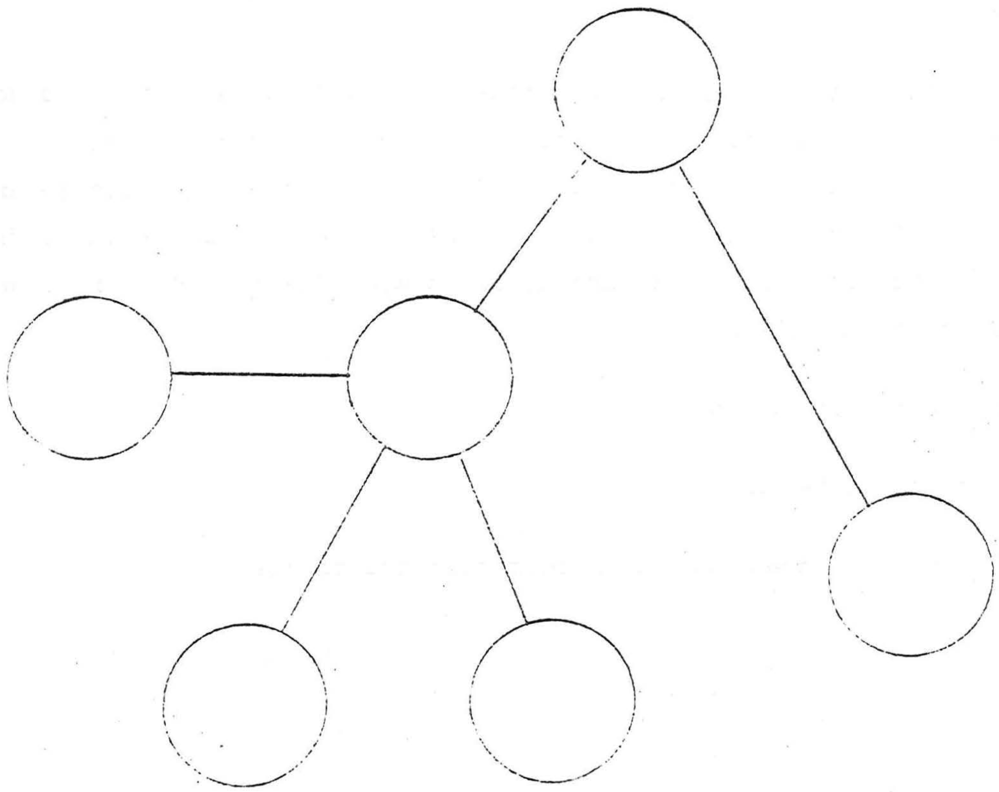


Figure 1a   Network Topology at Level 2

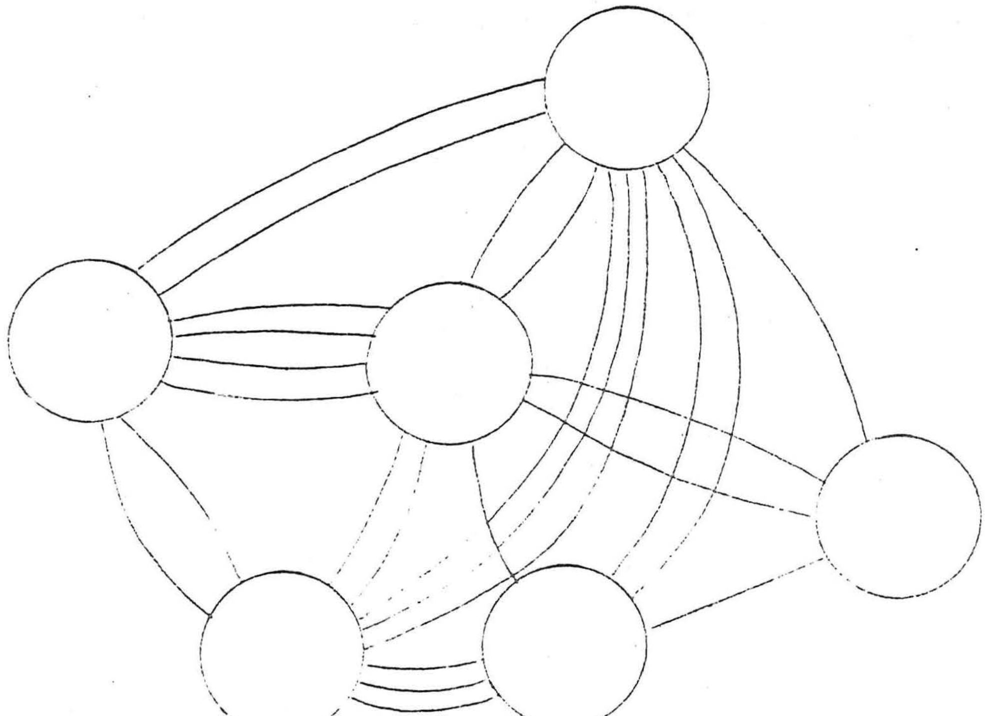
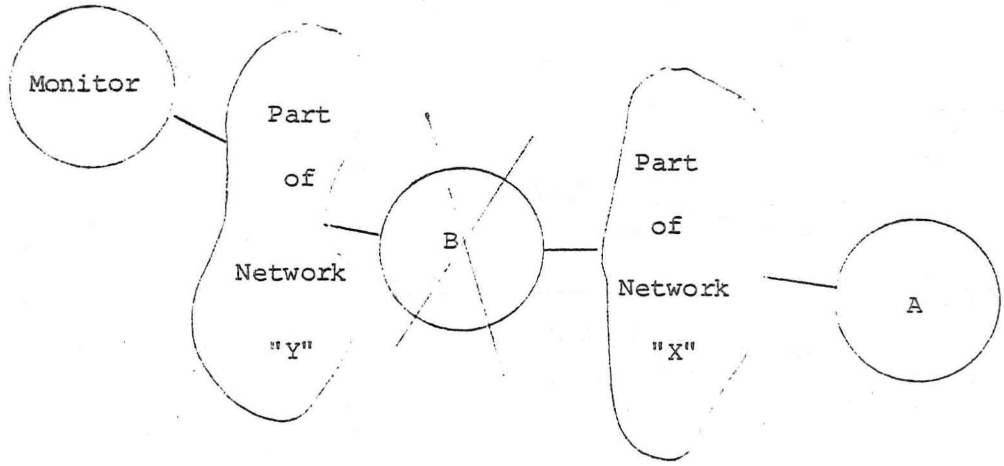


Figure 1b   Network Topology at Level 3



Status of	"Y"	-	ACTIVE
	B	-	BROKEN
	X) A)	-	UNKNOWN

Figure 2 Illustration of "Unknown" Component Status

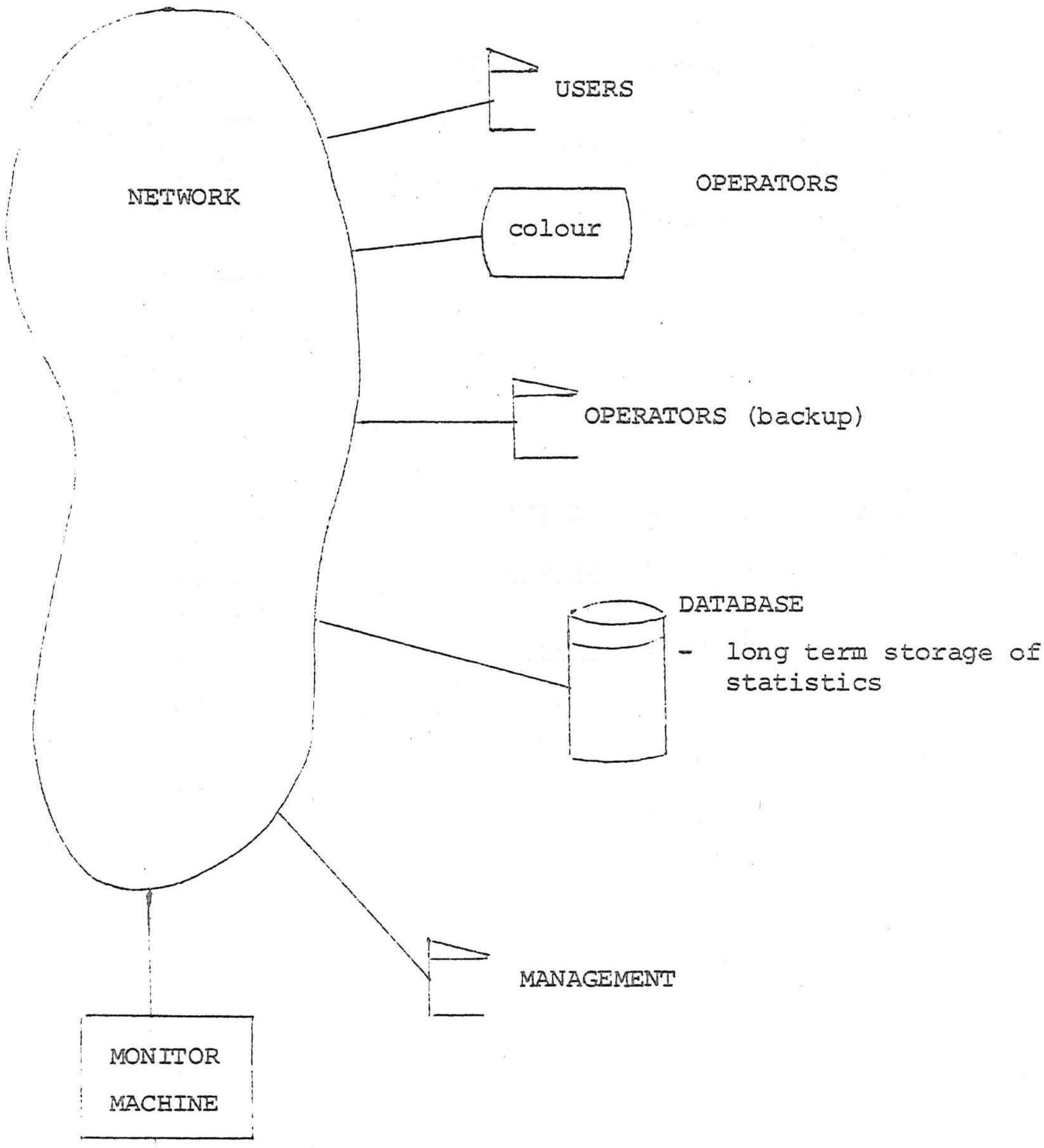


Figure 3 Logical Configuration of the Monitor System

PROVIDING AND MANAGING A LOCAL NETWORK SERVICE

Ian Dallas

University of Kent

PROVIDING AND MANAGING A LOCAL NETWORK SERVICE

I. N. Dallas, University of Kent at Canterbury

Introduction

The content of this paper is based on the experiences at the University of Kent, and can be considered to be a case study of how the Kent implementation of the Cambridge Ring local area network was brought into user service.

The Story so Far

The situation which existed at Kent was no doubt symptomatic of that on other campuses, with a rats' nest of cables, both within the Computer Laboratory, and campus-wide. Whenever two pieces of equipment needed to be connected together, then this required a new cable. As more and more terminals, departmental minis and micros were being installed, the situation was going to become steadily worse. The sensible way forward appeared to be the introduction of a local area network.

At that time an Ethernet type network seemed to be the best possibility, but there were several unknowns:

a) Could such a network be bought "off the shelf"?
b) What would be the cost?
c) How easy would such a network be to implement?

At this point, Networkshop 2 appeared, (April 1978), and revealed that a Cambridge Ring would be a much better prospect. It was cheaper than an Ethernet, and although nothing could be obtained "off the shelf", there appeared to be no real difficulty in building all the necessary equipment in house.

Progress reports on the actual building of the ring have been given at previous Networkshops, and it is sufficient to say here, that all went very smoothly.

With the ring built, and shown to be working, a decision was made to put it into user service, and its introduction was linked to a change in the ICL 2960 mainframe operating system from VME/K to EMAS. the Ring would play a similar role to the PDP-11/45 switch in the RCONET configuration.

Software development finished ahead of schedule, and on 2nd. January 1980, the system configuration changed from that shown in figure 1, to that shown in

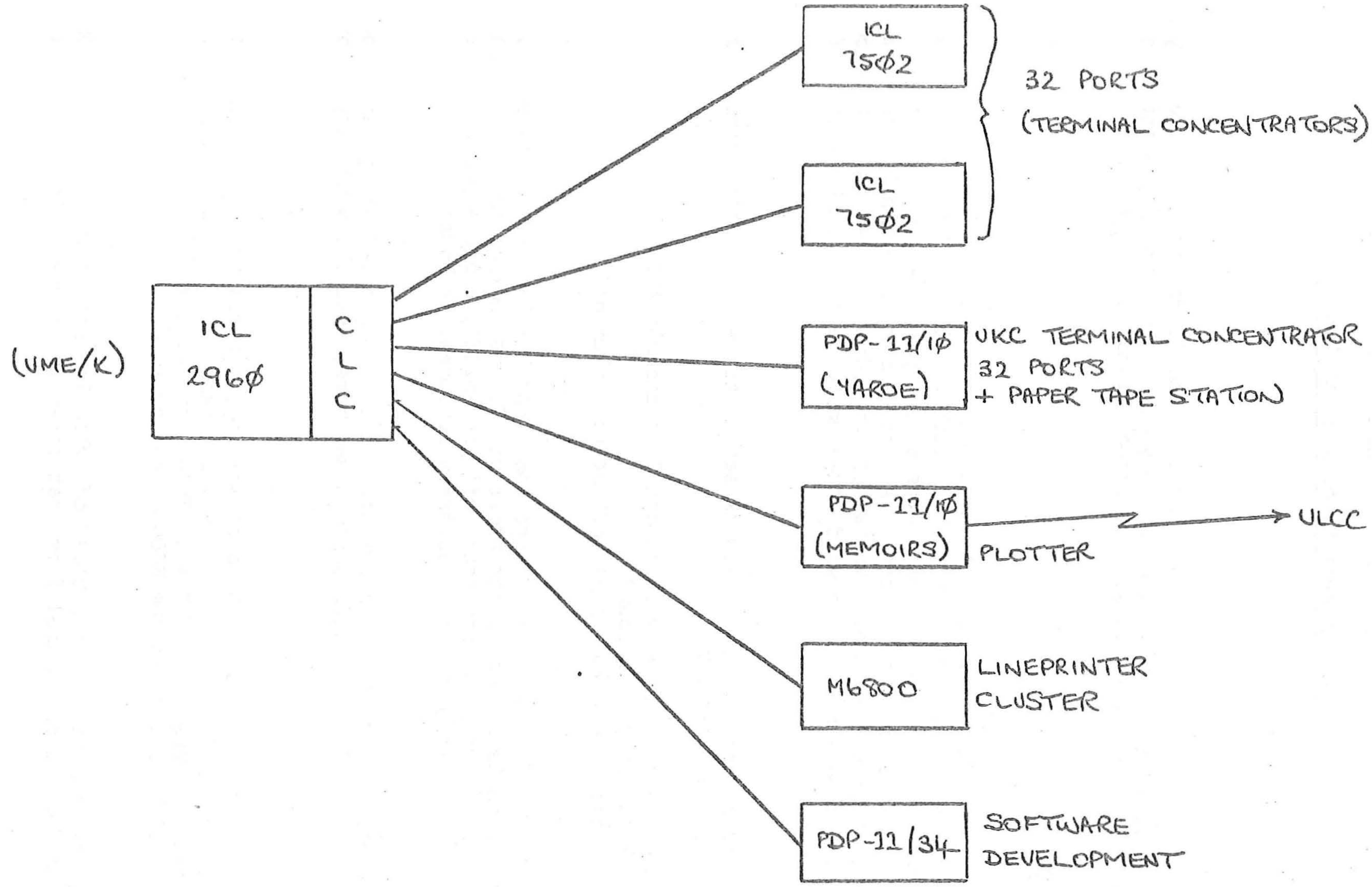


FIGURE 1 – KENT CONFIGURATION AUTUMN 1979

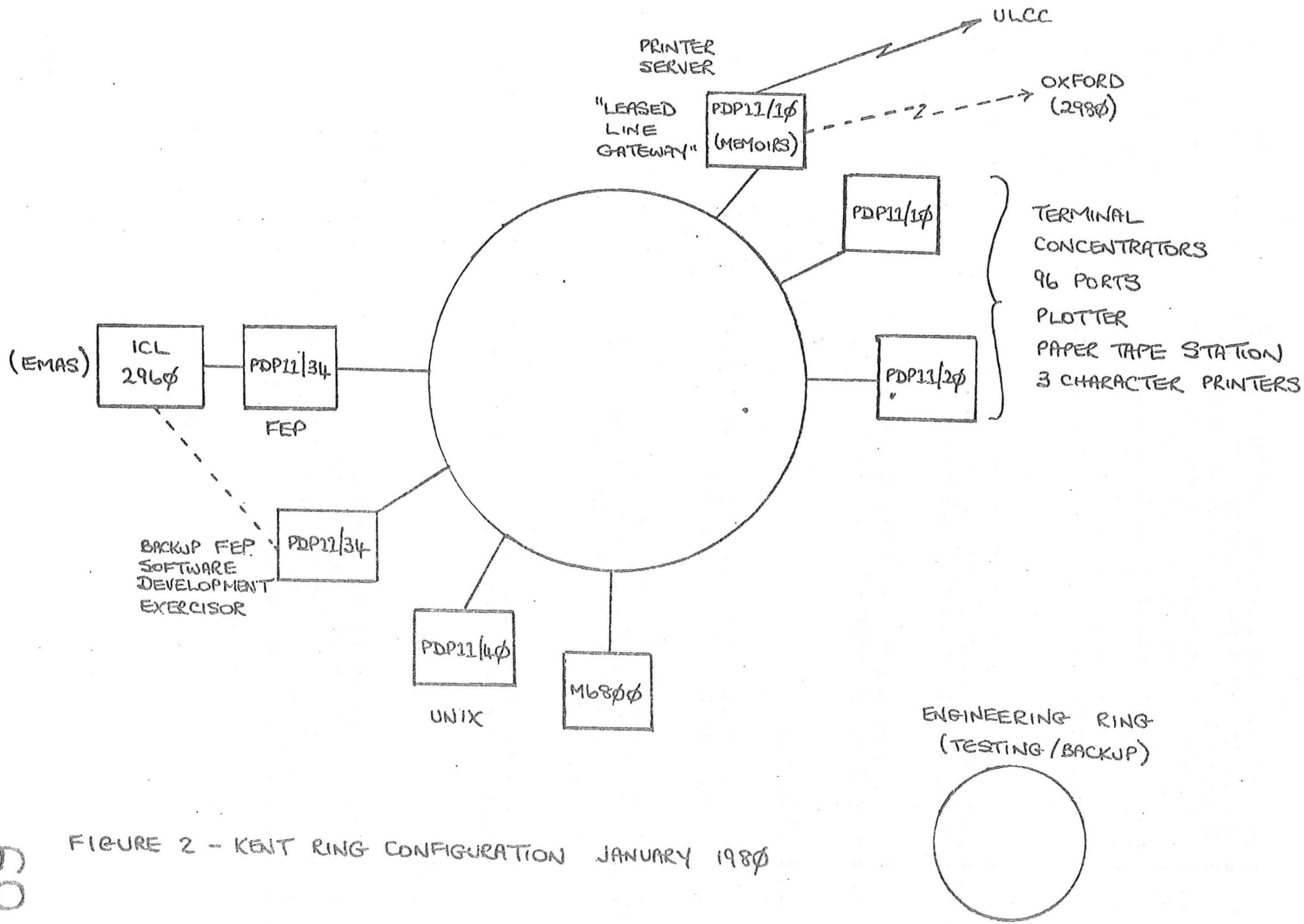


FIGURE 2 - KENT RING CONFIGURATION JANUARY 1980

figure 2.

There has now been three and a half months of operational experience, and the new service has had a good reception from the users, (although since the ring is invisible to them, this is more for EMAS). They have one terminal protocol, (not two slightly different), and some terminal speeds have been increased. The ring reliability has been good, with only 1 work station malfunction. Although there was no Error Logger on the ring at the time, the fault was picked up quickly through the software running in the PDP-11's at either end of a conversation. The work station module was replaced by a working spare always kept on hand, and subsequent investigation of the faulty station revealed a bit being dropped.

After initial teething problems with some of the ring drivers, and operational problems like the ring being powered off accidentally, the ring now provides a highly reliable and resilient system.

The error rate experienced on the ring has been in accordance with the rate predicted by Cambridge, namely 1 in 10 exp 11.

Hardware Expansion

Over the coming months, the ring service will be extended until by August 1980, the configuration will be as in figure 3.

The Error Logger and Name Server will be the next devices to be added. The former will enable better fault diagnosis, prediction and monitoring, whilst the latter will enable dynamic re-configuration of ring services, in the case of machine malfunction.

A new local host in the shape of a VAX 11/780 running UNIX will be added, as will another terminal concentrator, and 2 PASCAL microengines.

Protocols

In order to comply with the condition that the ring should become part of the user service by January 1980, it was decided to change the protocols used as little as possible. As a result, the protocol layer shown in figure 4, contained an amount of overkill, since there were 2 transport levels, the Byte Stream Protocol, (BSP), and the Node Standard Interface, (NSI).

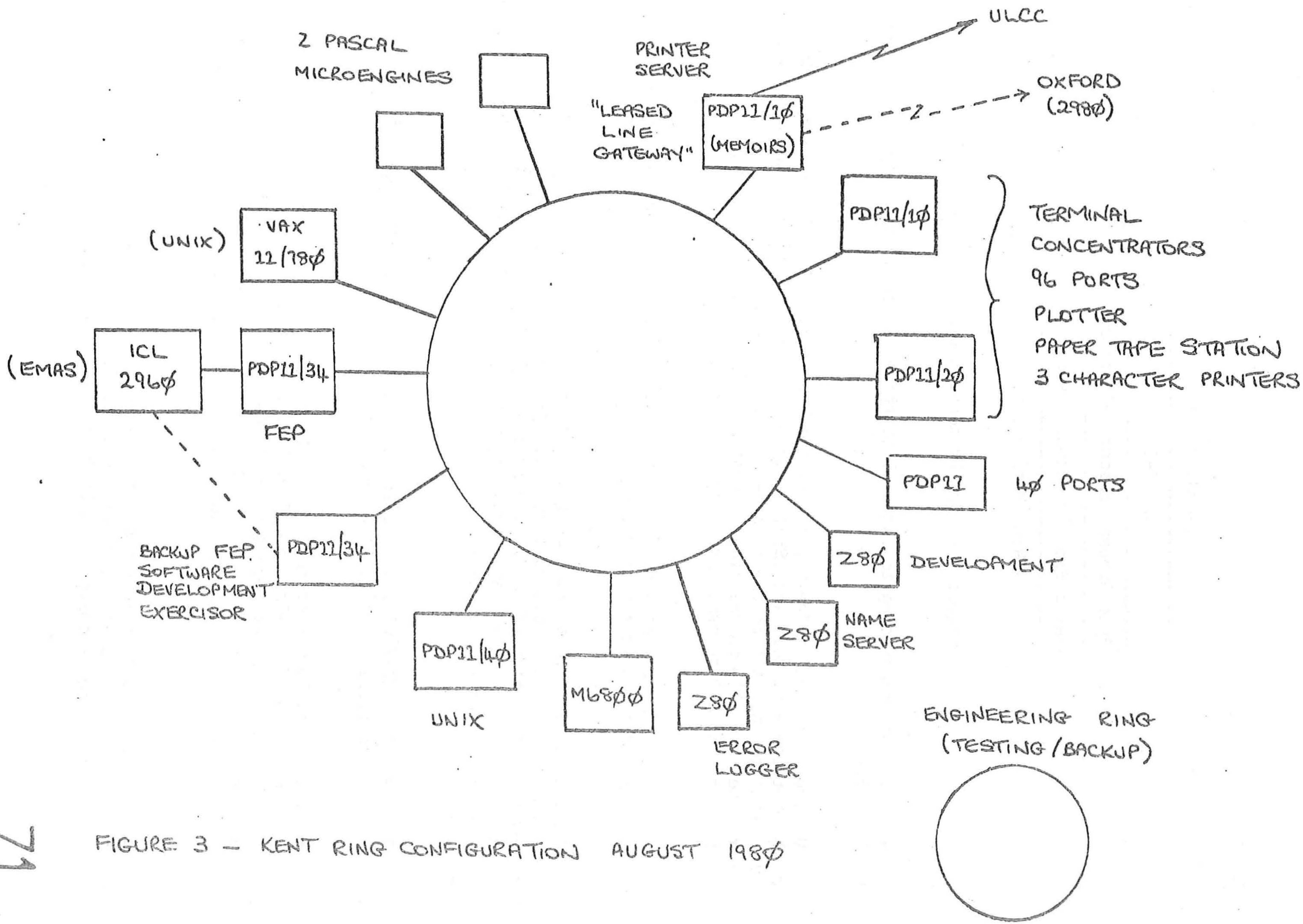


FIGURE 3 – KENT RING CONFIGURATION AUGUST 1980

ITP/RJE (RCO)

NSI (RCO)

Byte Stream Protocol (Cambridge)

Basic Block Protocol (Cambridge)

Ring Mini Packet

Figure 4 - Ring Protocol Layers

Work is now commencing to rationalise these layers, and to replace NSI in favour of BSP.

At the same time, a "new" protocol has been defined, called Transport Service Byte Stream Protocol, (or TSBSP). This is a definition of the additions which have to be made to BSP in order to get a realisation of the Yellow Book Transport Service {1}, on the ring. These additions are in fact very few in number, since the BSP commands are very close to the Yellow Book primitives.

TSBSP has been defined such that BSP as defined by the Systems Reasearch Group at Cambridge, forms a true subset.

The definition, (as at mid April 1980), is at the "Draft for Comment" stage, and has been sent out to interested parties. After all the comments have been received and processed, a final version of TSBSP will be produced and put forward as a Yellow Book Transport Service realisation for the Cambridge Ring, and it is hoped that it will have a similar status to Annex I of the Yellow Book.

The resulting TSBSP will form a solid base for all current and future higher level protocols, such as JTMP, FTP and TS29.

Because of the low error rate on the ring, there is no need for a protocol of the complexity of levels 2 and 3 of X25. BSP is defined completely in about 10 pages of A4, and all the necessary error correction is provided. Three months of operational experience has not shown up any flaws in the definition, and hence the low error rate high bandwidth network can be exploited to its fullest extent.

Due to the timescales involved, the VAX will initially use ITP as its interactive protocol, and whilst we are committed to FTP and JTMP, at the moment, we are uncertain about a change locally from ITP, (the RCO interactive proto-

col), to TS29.

There is no intention to stay with ITP forever, but one possibility under consideration is to wait for an internationally agreed Virtual Terminal Protocol to be defined before making a change.

The reasons for doing this are twofold. Firstly, users do not like change. ITP -> TS29 -> "VTP", means two changes, whilst ITP -> "VTP" means only one. Secondly, users want an upgrade of service, NOT a downgrade. In particular, interactive access to UNIX, requires single character transfers. Since TS29 is a line-at-a-time protocol it is inevitable that there would be howls of protest from users if such a feature were removed.

From an economic point of view, single character transfers should clearly be discouraged in wide area networks, but in a local area network like the ring, where there is plenty of bandwidth, which above all is free, then it should be exploited.

Extension of the Ring Service, and other Planned Developments

We believe that the ring at Kent can now be considered as a proven system, given that it is operating in a "data highway" mode.

The next step is to extend the ring service outside the Computing Laboratory, to allow other users to connect their equipment, and to ascertain if there are any consequential problems. We do not however envisage that this will be the case.

Two approaches are foreseen to this development, both of which are likely to be used. These are a Bridge, and the physical extension of the service ring.

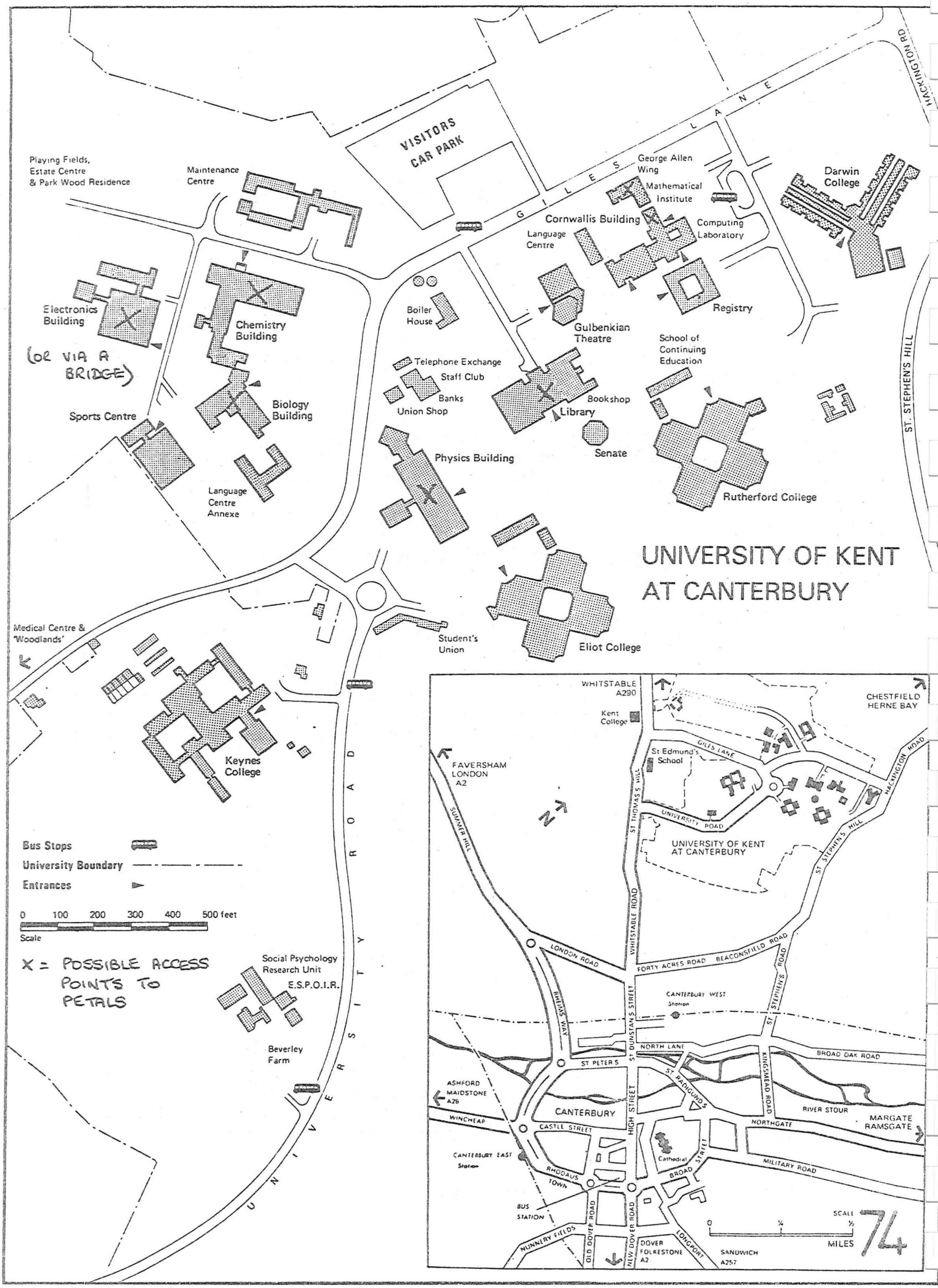
It is likely that there will be other rings on campus. These would be connected to the service ring by a bridge at the Transport Level. Any two users on such rings can communicate with each other in any way they want, but any user who wishes to communicate with the service ring will come under Computing Laboratory control, and since the bridge is at the Transport Level, must use TSBSP to so do.

The physical extension of the service ring will be via a series of "petals". These will be designed such that if there are an excessive number of errors, or a break in the ring, a petal can be isolated automatically via control from the error logger. Oxford University have already done some studies on such isolation, and we plan to work with them on this development.

A map of the University of Kent campus is shown in figure 5, with possible

University of Kent at Canterbury

FIGURE 5



74

access points to petals marked.

Either approach will mean that the service software must be resilient to rogue stations and/or users.

Before any equipment from outside the Computing Laboratory can use the service ring, it must first show that it can operate to the TSBSP exercisor. As its name implies this is an exercisor, and not a tester, (cf the Post Office PSS Development Aid), thus it can only be used as a first step. An example of a possible problem is as follows: A rogue station can send repeated, valid Opens say, but nothing else, which will tie-up the receiver in an equal number of timeout operations, (but the use of the Yellow Book Transport Service will mean that the identity of the rogue station will be available for action to be taken at the management level).

The service to the users will be extended by the addition of new terminal concentrators within departments. Some of the PDP-11's currently in use are of the order of 9 years old, and hence are beginning to show their age, indeed one was actually brought out of retirement! They are also somewhat expensive to use as concentrators, in terms of the cost per connected line. The new concentrators will be microprocessor based, using a Z80 or possibly an M6809.

The software in these concentrators will be based on TSBSP, and initially ITP, but it will be modular, so that the higher levels can be replaced in a straight forward manner at a later date. This software will be largely written in a "PL"-type high level language.

It is also hoped to connect the ring to PSS via a gateway. This will be at the transport level. If ITP continues to be used locally, there may be a protocol converter, to convert to and from ITP and TS29. Such a converter will be separate from the gateway, and no high level conversion will take place in the gateway itself.

Other longer term developments may include a File Server, Compiler Servers, and the conversion to an LSI ring.

Ring Management

Before the ring came into service, there was an operational worry about the throughput on the FEP PDP-11, since its interface to the ring worked on an interrupt per character basis, rather than DMA. This has however not proved to be critical, but DMA access to the ring is still being actively considered, and will probably be based on a Signetics 8X300 micro using work done at Cambridge, or possibly a DEC KMC11.

After the ring had been shown to our satisfaction to be viable, but before the decision to put it into service actually made, the mainframe interface was a

matter of major concern. At the time, the operating system on the ICL 2960 was VME/K, and besides the possible problem of actually interfacing to the operating system, there was an additional problem of the fastest communications line which the 2960 would support via a CLC was 9.6k baud, which gave quite a mismatch in speed with the ring. The majority of the problems simply disappeared with the decision to change to EMAS.

The final worry was about the suitability of the ageing PDP-11's. Again, this has not proved to be a problem, and the new microprocessor based terminal concentrators should overcome it completely.

Throughout the whole project, there have been minuted progress meetings and it is anticipated that these will continue until the project moves from a developmental to a normal state. These meetings monitor hardware and software progress, together with the state of the related documentation available. An important part of the meetings is to draw up priorities for the various tasks to be undertaken.

Management Issues Still to be Tackled

In the future some of the issues which we see as still needing to be tackled are:

(i) Control and Accounting Procedures - The ring is not heavily used, currently, but we see a need to have such procedures before the service is expanded much further, and the loading increases. A gateway would obviously handle such procedures in the wide area network context.

(ii) Fault finding and fixing procedures - We believe that these should be undertaken quickly, which naturally implies on-site fixing. It may well prove feasible to use our own unskilled (electronically) staff, (i.e. operators), to isolate a fault, and then replace the unit(s) for later repair, and restore the network service.

(iii) Arrangements for the maintenance, testing, commissioning and repair of equipment - At the moment, an engineering ring exists, but with limited facilities, which means that most testing and commissioning has to use the service ring, often during bookable "hands-on" sessions, when the technicians must compete with programmers for the available time. Plans must therefore be made to allow much greater use of the engineering ring.

(iv) Liaison with the University authorities and possibly the Post Office regarding the use of shared ducting for communications lines.

(v) Putting forward precise arrangements for connection, maintenance and monitoring of departmental minis - Use of the TSBSP exercisor has already been mentioned regarding initial connection, but there is a clear need to ensure that

the ring does not become clogged with erroneously transmitted packets, due to inefficient implementations of the protocols in such machines.

(vi) User services on a ring local area network, with particular reference to documentation, information services and help systems.

References

1. A Network Independent Transport Service, Prepared by Study Group Three of The Post Office PSS User Forum. February 1980. SG3/CP(80)2 1980-2-16.

LOCAL AREA NETWORKS : A SUMMARY REPORT

Ken Heard

Joint Network Team

LOCAL AREA NETWORKS - A SUMMARY REPORT

As many of you will know, we held a one day workshop on LANs at University College during February. I shall briefly report here the outcome of that meeting, and also review the potential local network products that are now being developed. I shall also say something about how they might be used, and of the need for higher level protocols and services to be used above the basic technology of the communications sub-network.

Purpose of a Local Area Network

A local area network is distinguished from any other type of network only by its dimensional extent. This limitation, usually to distances of less than a few kilometres but greater than a few centimetres, does however allow for the possibility of a significantly higher bandwidth than conventional telecommunications and can, in favourable circumstances, lead to considerable reductions in cost. In 1980, we tend to think of a LAN as about a kilometre long, moving bits at the rate of 1-10 megabits/second, and costing a few hundred (to a few thousand?) pounds to provide a computer-network connection. The rapid advances in technology could well see bit rates go up by a factor of 10 or more in the next few years, whilst costs should fall still further.

So, how can we take advantage of this style of network? There would seem currently to be three main purposes for using a LAN, deriving from the particular LAN characteristics of high performance, low cost and potentially pervasive nature:-

- economic connection of simple terminals
  (to service large numbers of users economically)
- interconnection of intelligent devices
  (to share specialised resources)
- distribution of processing functions
  (to achieve greater power, resilience, cost engineering)

The message I have received is that the attraction of LANs to our community is dominated by the apparent advantages for connection of user terminals, with an awareness of the benefits that could arise through the ability to share some types of resources, such as printers and filestores. The possibilities offered through distributed processing are generally reckoned to remain a matter for research.

In discussing the merits of local area networks, the February meeting recognised the importance of assessing the implications of the various proposed technologies (Rings, Ethers, X25 switches) on cost and performance, and of harmonising as far as possible the interfacing arrangements and higher level networking services that we used for local and wider area networks. Figure 1 lists the points for further study that were distilled from the earlier discussions. It is to be hoped that someone will do some work to provide the answers, or at least clearly identify the choices, to some of those problems. There are certainly some people who argue that the style of working over a local network will be so different to that over a long distance or wide area network (WAN) that there is little point in trying to force both systems to use the same access techniques and protocols. But it will clearly be a pity if all the work that has been put into providing for interworking over a WAN cannot be applied in the LAN case. Similarly, we must guard against the possibility that each LAN is so local that it has nothing in common with any other LAN. We need to understand and define the nature of LAN gateways, both between LANs and between LAN and WAN.

Standards must be defined for LANs, the benefits are too great to ignore. And why can't we allow for the capabilities (such as terminal handling, file transfer, etc) we have defined for WANs to work over LANs unmodified, using the concept of universal (standard) interface and protocol "plugs" as shown in Figure 2. There may be other, more sophisticated, higher performance or (may be) lower cost techniques that could be used with LANs, but it ought to be possible for them to coexist with what we already have.

LAN Products Survey

Figure 3 summarises the information we have been able to gather on current developments which could lead to productswithin the next year or so. The timescale for all potential offerings is remarkably similar - prototype product early in 1981 with general availability from mid-to-end of 1981. Local area networks everywhere are currently still in the R & D phase. Valuable experience is being gained, but they have not yet reached maturity.

There are, as I am sure you all know, three types of LAN based on very different techniques. The Cambridge Ring has been the subject of considerable study here in the UK, whilst the Ether is by far the more popular in the United States.

In fact, there seems to be rather little UK experience with Ethers, although there are 1 or 2 experimental systems within our community. I have included X25 packet switches for completeness, since their performance is not all that different from practical Rings and Ethers. More details on X25 switches will however be given in the next presentation.

The first thing to note about Cambridge Rings (CR) is that there are at least four different systems proposed, all incompatible at the detailed electrical and bit level. The "original" Cambridge Ring uses 38 bits for each mini-ring packet whereas the version to be based on the proposed LSI chip will have 40 bits. The Mark I Cambridge CR has traditionally been a DIY job, but it has now been taken up by a number of small firms. Consequently all installed CRs I know are different and will present an ongoing maintenance problem. The Mark II Cambridge CR awaits production of the chips, probably towards the end of the year. As a consequence, product manufacturers are currently unknown.

Two UK manufacturers are known to be developing Ring equipment as part of their product line, but to meet rather different markets. Racal Milgo are working on a CR, which will have 41 bit ring packets, to meet the traditional PACX need to interconnect devices with a standard V24 interface. The interface port throughput will therefore be restricted to less than 64K bps even though the Ring itself will be clocked at 5M bps bit rate.

The Linotype Paul system arises from their in-house need to conveniently interconnect a large number of terminals and other equipment involved in photo-typesetting. They have settled on 64 bit ring packets! This development has been undertaken top-down, from an understanding of the end user need, but is currently designed to interface directly with special purpose LP equipment. There is some possibility that LP will define a more general interface capability for a wider market. Both Racal and Linotype expect to have proven systems by the end of the year, with products available from mid 1981.

Much less is known about Ether developments, principally because the main thrust is taking place in the USA. Everybody knows that Xerox has devoted a lot of time and other resources to development of Ethernet: they have been working on it since 1975 and now have several inter-linked Ethernets in daily use. The Xerox system is however rather specialised in many ways (serving a large number of identical "Alto" minicomputers) and serves the limited objectives of a homogeneous corporation. It is also clear that Xerox could substantially influence the office equipment market, so we must certainly expect to meet, and interwork with, the Xerox Ethernet in the near future. Xerox are believed

to be considering releasing more details of their system soon, with a view to promulgating a standard local network, and hence widening their own opportunities in the market place.

Indeed we have recently heard rumours of a tie-up with a major micro manufacturer, which could lead to Ether interfaces and workstations on a chip. Again the timescales seem to be of the order of 1-1½ years.

Selection Criteria

The prospects for interconnection of computing equipment using one or other of the developing LAN technologies look good, but we are still, I fear, a long way from having solutions to all the problems that may arise in a service environment.

Clearly costs are important. Establishment and use of a LAN involves at least three distinct types of expenditure:

- unit cost of entry port to the network
- cost of interfacing 'user' equipment
- processing and other overheads involved in working over a LAN.

We should not forget the significance of the second two costs in relation to the attractive prices arising from technological developments used to provide the bare communications system.

Performance perceived by the user is important, but ought to be matched to real need. Overkill is expensive. At the same time, it does seem rather wasteful to build a LAN operating at 10M bps that can only offer a few K bps at the user access point.

We must allow for evolutionary extensibility: systems requirements always seem to grow, so we ought to be able to start with an initial system and enhance it progressively to meet increased demand. We must also be in a position to integrate a given system with others, through the use of controlled gateways or whatever.

Management aspects become crucial for any system operating in a service environment. Thus far, rather little effort has been devoted to provision of adequate management tools, inspite of the need to monitor the internal operation of particular LANs to ensure current operation. Distributed systems,

such as LANs, will require quite sophisticated management features if we are ever to make best use of their potential. Some of the features we can identify now are:

- maintenance diagnostics of incipient and actual failures
- resilience to common sources of failure
- control of equipment attachment
- control of user access
- accounting

KSH
18 July 1980

FIGURE 1

LAN Studies

1. Comparison of communications technologies
   - rings, ethers, centralised switches, etc

2. Provision of PSS compatible ports to LANs

3. Definition of alternative port interface(s)

4. Gateway considerations - LAN-LAN and LAN-WAN

5. Management aspects of LANs
   - control, accounting, maintenance, evolution

6. Terminal handling
   - styles of use, flexibility, maintenance, availability, cost

7. Sizing case study
   - what is really NEEDED

8. Detailed proposals for installation of a LAN on a given site
   - operational requirement - styles of use
traffic patterns
equipment to be connected
benefit
   - functional specification
   - development effort anticipated
   - timescales
   - cost

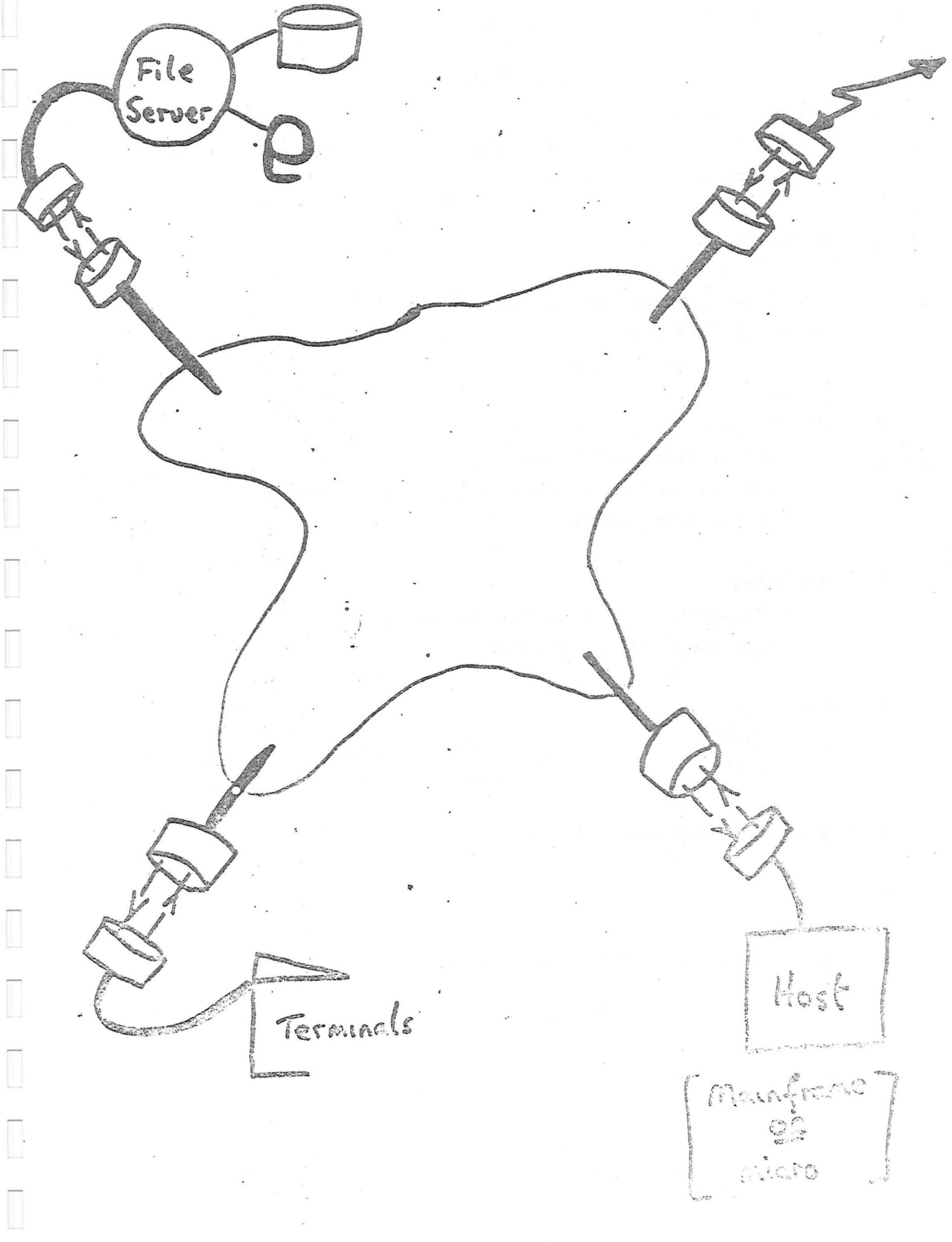


Figure 2

FIGURE 3

LAN Offerings

1. Cambridge Cambridge ring
   - Mark I 38 bit pkts (various small producers)
   - Mark II 40 bit pkts (?)

2. Racal Milgo CR
   - 5M bps data rate on ring
   - V24 interface ports to 64K bps
   - PACX like operation
   - 41 bit ring packets

3. Linotype Paul CR
   - 5M bps data rate on ring
   - peculiar 15 wire interface (to LP requirements)
   - 64 bit ring packets

4. Xerox Ether
   - specialised to meet limited objective
   - domination? of office market?

5. Three River
   - ether?
   - ring?

   commercial "tie-up" with other (computer) supplier(s)

6. Major micro manufacturer - Ether

7.

8. Centralised packet switches based on X25

9.

ETC, ETC, ETC ....

INTERIM AND LONGER TERM SOLUTIONS FOR TERMINAL HANDLING

Rick Blake

University of Essex

REQUIREMENTS FOR CHARACTER TERMINAL HANDLING

Many sites are expressing a pressing need for the ability for terminals to switch and contend for access to a number of hosts. Some sites have developed interim solutions in this area but many of these will need replacement before too long while many other sites have nothing at all. In order to assess what general products are needed in this area, it is first necessary to assess what the requirements are.

In this paper, we are assuming that the term "character terminal" encompasses the set of character mode devices that are commonly connected to a computer by use of asynchronous communication lines, with particular emphasis on keyboard devices. No attempt has been made to consider the peculiar problems posed by remotely refreshed graphical devices or high speed printers, both being devices that may demand extremely high instantaneous throughput rates over asynchronous lines.

This paper attempts to give a rough idea of the quantitative requirements for terminal handling on the average campus. It does not go into the general operational requirement but rather attempts to quantify such things as number of terminals and throughput.

Some of the figures given below come from a questionnaire recently sent to the universities and Research Councils. Some come from particular inquiries while others are guesses. More throughput measurements are badly needed.

Number of terminals

These numbers are expressed as a range. Average represents the mean of all sites. Low represents the situation on a small campus leaving out those with extremely little terminal use while high represents a large terminal user site excluding only the most extreme examples.

	Now	1985
	Low - Av. - High	Low - Av. - High
No. of terminals	50 - 120 - 350†	80 - 240 - 500†
Max. No. of active terminals	40 - 65 - 200†	60 - 115 - 300*

†information from 34 returned terminal handling questionnaires

*guesses

It is of interest to note that, although the current contention ratio between the average numbers of active and existing terminals is slightly less than 2:1, and the predicted average value in 1985 is slightly greater than 2:1, information from sites with a great deal of experience of terminal handling indicates that a successful contention ratio is more of the order of 3:1.

Terminal throughput examples

These are specific measured examples intended to give a general idea. More details are available on request.

From host viewpoint

1. Daresbury TSO during busiest hour with 50 users, average terminal speed 4800 bps: Av.: 600 ch/sec
26 lines/sec

2. Essex TOPS10 with 40 users, average terminal speed 2400 bps:

XMT: Av.: 350 ch/sec
14 lines/sec

RCV: Av.: 15-120 ch/sec
$\frac{1}{2}\rightarrow 4$ lines/sec

Note: the high RCV figure is due to intercomputer traffic.

3. Cambridge PHOENIX with 80 users, average terminal speed 1200 bps.

XMT: Av.: 750 ch/sec
31 lines/sec

RCV: Av.: 75 ch/sec
3 lines/sec.

From terminal viewpoint

1. Using Daresbury TSO, av. user on 4800 bps terminal works at 12 chars/sec, $\frac{1}{2}$ line/sec

2. Using Exeter System 4, av. user on 2400 bps terminal works at 3 chars/sec, 1/8 lines/sec

It will be noted, firstly, that the measured figures are enormously smaller than the theoretical maximum throughput rate (typically less than 5%) and secondly that the vast majority of character traffic is in the output direction. These figures have been used to generate the example solution (for a hypothetical site) outlined in the next section; the assumption has been made that a communications line from a multiplexor will provide adequate response at times of instantaneous peak loading if it is rated at a speed of approximately 3 times the average throughput. The hypothetical site is assumed to have a Local Area Network (of an unspecified type); the throughput requirements of the network are also indicated.

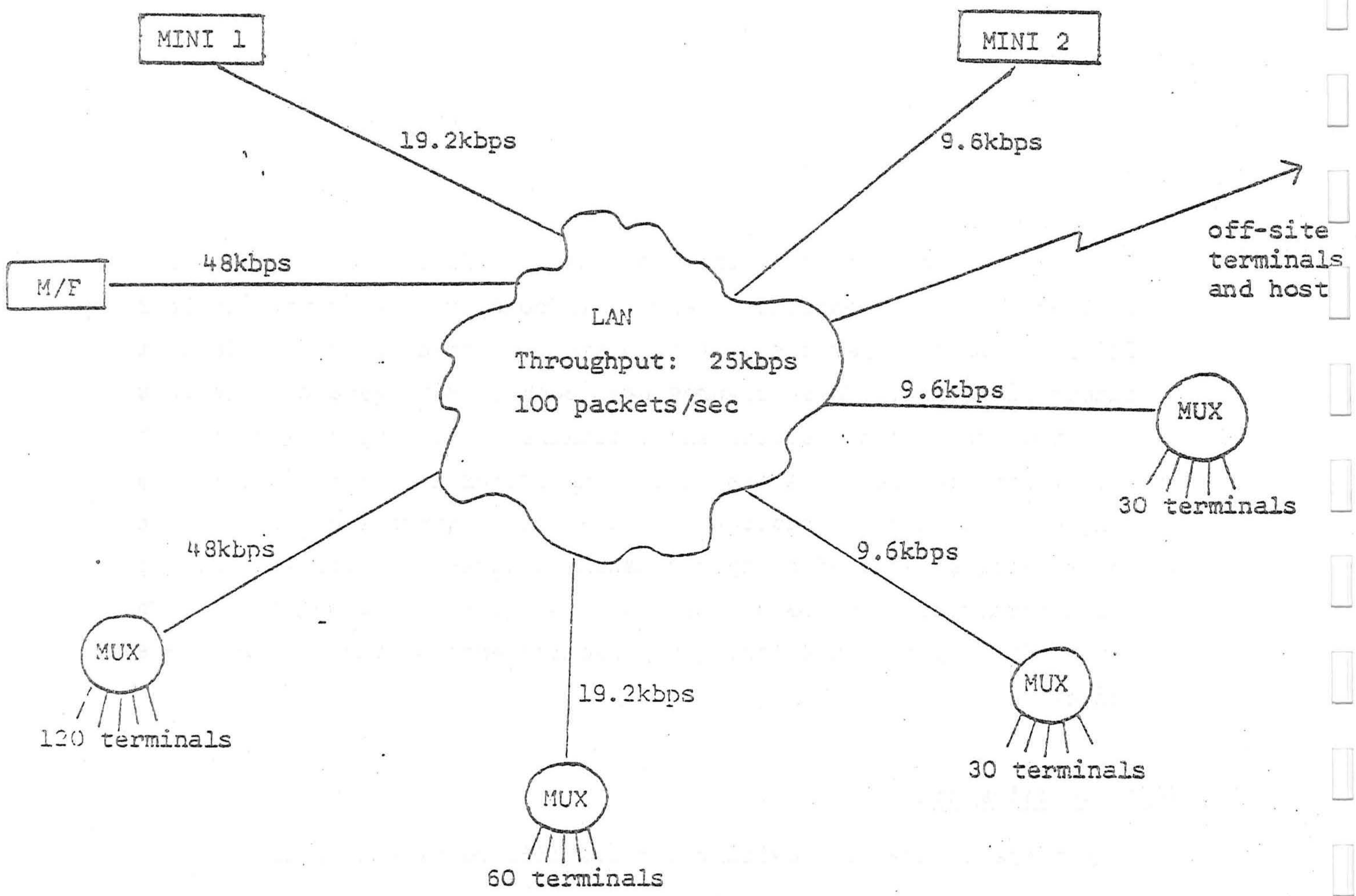
A hypothetical site

Assume an average site is providing for its 1985 requirements of:

240 terminals
115 max. no. active at any one time
1 mainframe host handling 83 terminals with busiest hour
av. throughput 1200 chars/sec, 48 lines/sec
1 mini with 24 terminals with busiest hour
av. throughput 500 chars/sec, 20 lines/sec
1 mini with 8 terminals with busiest hour
av. throughput 300 chars/sec, 12 lines/sec

Total throughput required from above is 2000 chars/sec or 80 packets/sec assuming one line per packet.

A solution might look like:



It should be noted that the line speeds and throughput figures given are merely those considered to be sufficient to ensure good terminal performance; it is probable that the line speeds could be lowered, at the expense of occasional response time degradation, whilst the LAN will almost certainly be capable of higher throughput rates than those quoted.

The structure employed on an actual site will, of course, depend significantly upon its geographical distribution and will be chosen using economic considerations.

Dr. R.G. Blake
Dr. B.J. Charles

Draft specification for a triple-X PAD

1. Introduction

The growing diversity of computing services available to users within the University and Research Council community and the geographical distribution of the sites which offer these services has led to a situation in which sets of terminals within university departments need to be able to connect to more than one host computer. Traditionally circuit switching based on the PSTN has been used, supplemented in cases where leased lines are used by contention and/or multiplexing hardware. The adoption of other technologies - particularly X.25-based packet switching - requires the development of a number of new switching and concentrating devices, one of which is the PAD described here.

While this document describes a PAD (Packet Assembly/Disassembly) facility for asynchronous terminals, it should be borne in mind that a number of devices performing a range of network-related functions will also be required. For example, devices to accept hardcopy output from a number of hosts will be needed, as will the networking analogue of traditional Remote Job Entry terminals. In interpreting this specification, therefore, the need for devices of these types (and hybrid devices combining a number of such functions) should not be forgotten, and it is recommended that the software for the PAD should primarily provide X.25 (and Transport Service) support, with the PAD function being considered as an "application" which uses this basic package.

The PADs will be used as concentrators, allowing a number of asynchronous terminals to access a mainframe through a single synchronous connection, and also to provide switching facilites by providing access (through the medium of X.25 calls) to a number of different hosts either on- or off-site. The basic design should be flexible, especially in its ability to be readily configured to meet the requirements of different sites in such matters as number and speed of terminals, synchronous line data rates, local addressing conventions, etc.

It is expected that the program for the PAD will run stand-alone, but if the design is based on a processor for which a widely-used operating system exists then a separate version of the program which runs under that system would be highly desirable.

2. Specification of the PAD

2.1 Function

This device is required to conform to CCITT recommendations X.3, X.28 and X.29 and to the recommendations of the Character Terminal Working Group of PO Study Group 3 (the "Green Book"). A number of parameters, options, etc. are either not yet defined within these recommendations, or

are left as implementation decisions. It is intended, however, that the PAD defined here should functionally be as compatible as possible with the PAD to be provided by the Post Office as part of its PSS. Accordingly, the Post Office decisions as expressed in Part IV of the PSS User Guide will be adopted unless there are strong reasons against this.

2.2 Capacity

It is already clear that PADs covering a wide range of capacities will be required. At the low end a unit capable of handling 8 asynchronous terminals at speeds up to 9.6 kbps with synchronous "DCE" connections up to 9.6 kbps is required, while at the upper end in the order of 100 asynchronous ports and one or more synchronous lines operating at 48 kbps must be supported. Whether the larger configurations are built up from a number of the smaller units or are based on an entirely different design will be determined mainly by economic factors - it is important that the "cost per port" is minimised. The provision for multiple DCE links is to allow the PAD to be directly connected to more that one local host (or to a mixture of hosts and networks). This implies that the PAD must be able to route data down one or other of the links according to local addressing conventions, and that it should be capable of acting either as a DTE or a DCE (in such matters as choice of logical channel numbers, resolution of call collision, etc.), possibly even operating in different modes simultaneously on different lines.

2.3 Monitoring and control facilites

Optional control and monitoring of the operation of the PAD must be provided for, both directly, through a locally attached terminal and remotely, by means of a call conforming to X.29. The monitoring facilities should be capable of logging the following:

1. At call set-up:

   Called DTE address
   Calling terminal identity
   Calling NUI (where appropriate)

2. At call termination:

   Called DTE address
   Calling terminal identity
   Calling NUI (where appropriate)
   Number of bytes, packets and "chargeable" segments read and written
   Call duration.

(Note: "chargeable" segments are segments for which the Post Office would have charged if this call had been made through a public PAD.)

3. For each synchronous line, at specifiable

intervals, the number of bytes and blocks read and written, and itemised error counts, all since the last report.

4. Immediate notification of catastrophic line errors, or of error rates in excess of a user-defined threshold.

All monitoring information should be time-stamped.

The following control facilities will be required:

1. Ability to establish monitoring intervals and error rate thresholds

2. Ability to add and delete NUIs

3. Ability to establish access controls based on NUI/called DTE address combinations, possibly using installation-defined vetting procedures.

4. Ability to enable and disable individual synchronous and asynchronous ports.

2.4 Operation

The PAD must be capable of running with a minimum of operator intervention. Ideally it should have no operator controls other than an on/off switch and, perhaps, a restart button. It is expected that the program will be held in read-only memory within the PAD, and that some form of watchdog facility will be incorporated to restart the program automatically in the event of a failure. Wherever possible, relevant data should be logged at the time of the failure to permit its rapid diagnosis and rectification.

Versions of the software modified to run under an operating system should include appropriate commands to simulate the effects of the off and restart switches. Additional commands or procedures may also be necessary to associate logical ports with actual interfaces on the system.

2.5 Availability

Since access to all interactive computing services (wherever provided) is likely eventually to be routed through PADs it is essential that these devices should offer a high degree of reliability, exhibited as a mean time between failures (MTBF) in excess of 2500 hours, and an availability better than 93% in any one week and 99.5% in any year (assuming the equipment to be in use for 24 hours per day). Availability is here taken to mean the time for which the device was actually available to users, expressed as a percentage of the time that it was scheduled to be

available (ie., excluding periods of routine maintenance, power cuts, etc.).

3. Enhancements

3.1 Mnemonic addressing

In addition to the normal mechanism for entering the network address of a service to be called from a PAD, extensions are required to allow remote processes to be identified by means of a mnemonic addressing string rather than the full network address. The PAD would be required to replace the mnemonic address with the network address. Since these address strings will in the main be associated with timesharing services on remote hosts, it should be possible to associate a default (extended) terminal profile with each mnemonic address.

The Reverse PAD

In the following it is assumed that the bulk of interactive terminal traffic - certainly between sites - will be observing the triple-X family of protocols (X.3, X.28, X.29, TS29) within the next few years. This trend will have implications for both the users and providers of interactive services.

For terminals it requires the development of PAD facilities, i.e. the ability to accept individual characters from a keyboard, assemble them into packets and forward them to the host at an appropriate time (e.g. on receipt of a carriage return character) and to receive packets from the host and output the data to the terminal. The PAD may perform a number of additional functions - e.g. echoing characters from full duplex terminals.

At the other end of a connection the ideal situation would be to have a host which fully supports X.25 and the triple-X protocols. While there are signs that such facilities will be available with at least some members of the next generation of systems, there already exists a multitude of hosts which are incapable of digesting the new protocols as they stand, and would require extensive modifications in order to be able to do so. If, as seems essential, such systems are required to provide triple-X based services then a solution (or range of solutions) to this problem must be developed.

Clearly the ideal would be to develop triple-X support for the existing equipment. This would of course consume a large amount of scarce effort, since it would have to cover the complete range of machine types and operating systems now in use, adapted as necessary to take account of local modifications. Timescales would vary according to the complexity of the system and the availability of manpower with the appropriate skills. In a number of cases the benefits derived (namely the ability to provide access for a small number of external users) would be totally swamped by the trauma the conversion process would cause to existing local users.

An alternative approach is to consider the provision of a unit which will stand between the triple-X network and the existing system, demultiplexing the triple-X data streams, and presenting them to the host over separate asynchronous connections as if they were being generated by local (or dial-up) terminals. Such a device (a reverse PAD) offers a number of advantages (in the appropriate environment).

Firstly, it allows an existing host to accept a modest amount of traffic over a triple-X network with no software or hardware modifications with the possible exception of a few additional asynchronous interfaces. Secondly, at such time as the host can be conveniently converted to support triple-X, or is replaced by a system with this capability, the reverse PAD can be reprogrammed to act as a PAD for the existing on-site terminals (the off-site connections now being handled directly by the host) giving them the potential to access new services over the external network. Thirdly, the effort involved in defining and implementing a reverse PAD should be considerably smaller than that required to implement triple-X on the hosts, and the timescales should be correspondingly shorter.

The major difficulty in the definition of a reverse PAD lies in the variety of protocols used between host computers and asynchronous terminals. Differences can be expected in such areas as

1. Establishing and breaking the connection to the host.

2. Breaking of a connection by the host.

3. Character echoing for full duplex operation.

4. Recognition of "forwarding conditions" for messages from the host.

5. Character set mapping.

6. Parity generation/checking.

7. Resolution of conflicts between simultaneous input and output operations.

8. Mechanism by which interrupt signals should be notified to the host (if at all).

Hopefully it would be possible to define these protocols for a relatively small number of widely-used systems, and to cope with most other systems by some form of "port profile" (by analogy with the triple-X terminal profile) with a set of parameters to define the way in which the reverse PAD - host link should be handled. There may remain a few pathological cases for which special handlers would have to be written - or a different approach adopted.

Two approaches to solving the problems of connecting existing hosts to triple-X networks have been described here - probably there are others. You are invited to consider

1) whether you will have a need to accept triple-X interactive traffic

2) whether your existing system is capable of this

3) if not, whether the first solution described above is a) desirable, b) feasible or whether a reverse PAD would adequately stave off the problem.

Feedback on all these points is vital if programs of work are to be put in hand.

H. C. Kirkman
University of London Computer Centre
20 Guilford Street
London WC1N 1DZ

28th February 1980

Draft specification for a triple-X PAD

1. Introduction

The growing diversity of computing services available to users within the University and Research Council community and the geographical distribution of the sites which offer these services has led to a situation in which sets of terminals within university departments need to be able to connect to more than one host computer. Traditionally circuit switching based on the PSTN has been used, supplemented in cases where leased lines are used by contention and/or multiplexing hardware. The adoption of other technologies - particularly X.25-based packet switching - requires the development of a number of new switching and concentrating devices, one of which is the PAD described here.

While this document describes a PAD (Packet Assembly/Disassembly) facility for asynchronous terminals, it should be borne in mind that a number of devices performing a range of network-related functions will also be required. For example, devices to accept hardcopy output from a number of hosts will be needed, as will the networking analogue of traditional Remote Job Entry terminals. In interpreting this specification, therefore, the need for devices of these types (and hybrid devices combining a number of such functions) should not be forgotton and it is recommended that the software for the PAD should primarily provide X.25 (and Transport Service) support, with the PAD function being considered as an "application" which uses this basic package.

The PADs will be used as concentrators, allowing a number of asynchronous terminals to access a mainframe through a single synchronous connection, and also to provide switching facilities by providing access (through the medium of X.25 calls) to a number of different hosts either on- or off-site. The basic design should be flexible, especially in its ability to be readily configured to meet the requirements of different sites in such matters as number and speed of terminals, synchronous line data rates, local addressing conventions, etc.

It is expected that the program for the PAD will run stand-alone, in order to avoid licence fees for manufacturer's operating systems, but if the design is based on a processor for which a widely-used operating system exists then a separate version of the program which runs under than system would be highly desirable.

2. Specification of the PAD

2.1 Function

This device is required to conform to CCITT recommendations X.3, X.28 and X.29, and specifically that subset defined by the Character Terminal Working Group of PO Study Group 3. A number of parameters, options, etc. are either not yet defined within these recommendations, or are left as implementation decisions. It is intended, however, that the PAD defined here should functionally be as compatible as possible with the PAD to be provided by the Post Office as part of its PSS. Accordingly, the Post Office decisions will be adopted unless there are strong reasons against this.

2.2 Capacity

The design should be capable of supporting a maximum of 32 asynchronous terminals at speeds up to 9.6 kbps, although most configurations would be expected to handle no more than 8 terminals. There should be one or more "DCE" connections by synchronous lines (using X.25 LAP-B) at up to 9.6 kbps. The provision for multiple DCE links is to allow the PAD to be directly connected to more than one local host (or to a mixture of hosts and networks). This implies that the PAD must be able to route data down one or other of the links according to local addressing conventions, and that it should be capable of acting either as a DTE or a DCE (in such matters as choice of logical channel numbers, resolution of call collision, etc.), possibly even operating in different modes simultaneously on different lines.

2.3 Monitoring and control facilities

Optional control and monitoring of the operation of the PAD must be provided for, both directly, through a locally attached terminal and remotely, by means of a call conforming to X.29. The monitoring facilites should be capable of logging the following:

1. At call set-up:

   Called DTE address
   Calling terminal identity
   Calling NUI (where appropriate)

2. At call termination:

   Called DTE address
   Calling terminal identity
   Calling NUI (where appropriate)
   Number of bytes, packets and "chargeable" segments read and written
   Call duration.

(Note: "chargeable" segments means segments for which the Post Office would have charged if this call had been made through a public PAD.)

3. For each synchronous line, at specifiable intervals, the number of bytes and blocks read and written, and itemised error counts, all since the last report.

4. Immediate notification of catastropic line errors, or of error rates in excess of a user-defined threshold.

All monitoring information should be time-stamped.

The following control facilities will be required:

1. Ability to establish monitoring intervals and error rate thresholds.
2. Ability to add and delete NUIs.
3. Ability to establish access controls based on NUI/called DTE address combinations, possibly using installation-defined vetting procedures.
4. Ability to enable and disable individual synchronous and asynchronous ports.

2.4 Operation

The PAD must be capable of running with a minimum of operator intervention. Ideally it should have no operator controls other than an on/off switch and, perhaps, a restart button. It is expected that the program will be held in read-only memory within the PAD, and that some form of watchdog facility will be incorporated to restart the program automatically in the event of a failure. Wherever possible, relevant data should be logged at the time of the failure to permit its rapid diagnosis and rectification.

Versions of the software modified to run under an operating system should include appropriate commands to simulate the effects of the off and restart switches. Additional commands or procedures may also be necessary to associate logical ports with actual interfaces on the system.

2.5 Availability

Since access to all interactive computing services (wherever these are situated) is likely eventually to be routed through PADs it is essential that these devices should offer a high degree of reliability, exhibited as a mean time between failures (MTBF) in excess of 2500 hours, and an availability better than 93% in any one week and 99.5% in any year (assuming the equipment to be in use for 24 hours per day). Availability is here taken to mean the time for which the device was actually available to users, expressed as a percentage of the time that it was scheduled to be available (i.e. excluding periods of routine maintenance, power cuts, etc.).

3. Enhancements

3.1 Formatting and editing

The procedures defined by the recommendation X.28 for communication between the asynchronous terminal and the PAD are not particularly well-suited to use within a time-sharing environment, especially when compared with the sorts of facilities made available to directly-connected terminals on most modern computing systems. For example, most systems send an automatic line feed (possibly with NULs inserted) in response to a carriage return from the terminal. Control of carriage movement in this way through a network (or series of networks) is unlikely to be satisfactory in view of the transit delays which may be experienced. There is therefore a case for a PAD which has a range of such formatting functions built in. There is, additionally, a need for a range of local editing facilities which are line- rather than message-oriented (as in the PO PAD).

It is, of course, essential that the data flow to and from such a PAD should exactly match that from a standard X.3, X.28-conforming PAD. The use of 3X in conjunction with existing time-sharing systems will require the definition of sets of procedures to be followed by terminal users wishing to communicate with these systems. It is desirable that the PAD should contain extensions which allow these procedures to be partly implemented within the PAD itself. Examples of such procedures and extensions are defined in the report of the Character Terminal Working Party Group of PO Study Group 3.

3.2 Mnemonic addressing

The PO will offer "direct calling" as an option on their PADs. This will allow a character terminal to establish a call to a pre-arranged address without the need to specify that address when the call is requested. While this facility may still require to be present in the Universities' PAD (for compatability with the PO), extensions are also required to allow remote processes to be

identified by means of a mnemonic addressing string rather than the full network address. The PAD would be required to replace the mnemonic address with the network address. Since these address strings will in the main be associated with timesharing services on remote hosts, it should be possible to associate a default (extended) terminal profile with each mnemonic address.

The following tables indicate the terminal traffic through our DN87 terminal concentrator (front-ending a DEC KL1Ø). The "RCV side" measurements refer to characters received from the terminals, whilst the "XMT side" measurements refer to characters transmitted to the terminals, the average speed of which was 2400 baud. For the morning period, when no terminal numbers were recorded, the average number of terminals was 5-7.

DN87 RCV Side

Results on 2 APR 1980

time	#	diff	time in secs	cps
1152:00	345	-	-	-
1155:00	801	456	180	2.5
1200:00	1725	924	300	3.1
1205:00	2623	898	300	3.0
1210:00	3965	1342	300	4.5
1219:00	5789	1824	540	3.4
1233:00	6187	398	240	1.7

DN37 RCV Side   Results on 2 APR 1980

time	#	diff	time in secs	cps	TTYs
1455:00	43004	-	-	-	17
1500:00	44904	1900	300	6.3	17
1510:00	49650	4746	600	7.9	17
1518:00	53419	3769	480	7.8	19
1525:00	57337	3918	420	9.3	21
1535:00	61765	4428	600	7.4	20
1540:00	64808	3043	300	10.1	19
1547:00	68963	4155	420	9.9	16
1553:00	73555	4592	660	7.0	17
1609:00	78386	4831	660	7.3	20
1615:00	81987	3601	360	10.0	20

DN87 XMT Side

Results on 2 APR 1980

time	#	diff	time in secs	cps
1152:00	11603	–	–	–
1155:15	33000	21397	180	119
1200:15	57048	24048	300	80
1205:15	69153	12105	300	40
1210:15	95626	26473	300	88
1219:15	110994	15368	540	28
1223:15	125704	14710	240	61

DN87 XMT Side

Results on 2 APR 1980

time	#	diff	time in secs	cps	TTYs
1455:15	16655	-	-	-	17
1500:15	49153	32498	300	108	17
1510:15	120648	71495	600	119	17
1518:15	190884	70236	480	146	19
1525:15	260782	69893	420	166	21
1535:15	321112	60330	600	101	20
1540:15	370490	49378	300	165	19
1547:15	427013	56521	420	135	16
1557:15	499539	72526	600	110	17
1609:15	598566	99027	720	138	20
1615:15	628346	29780	360	83	20

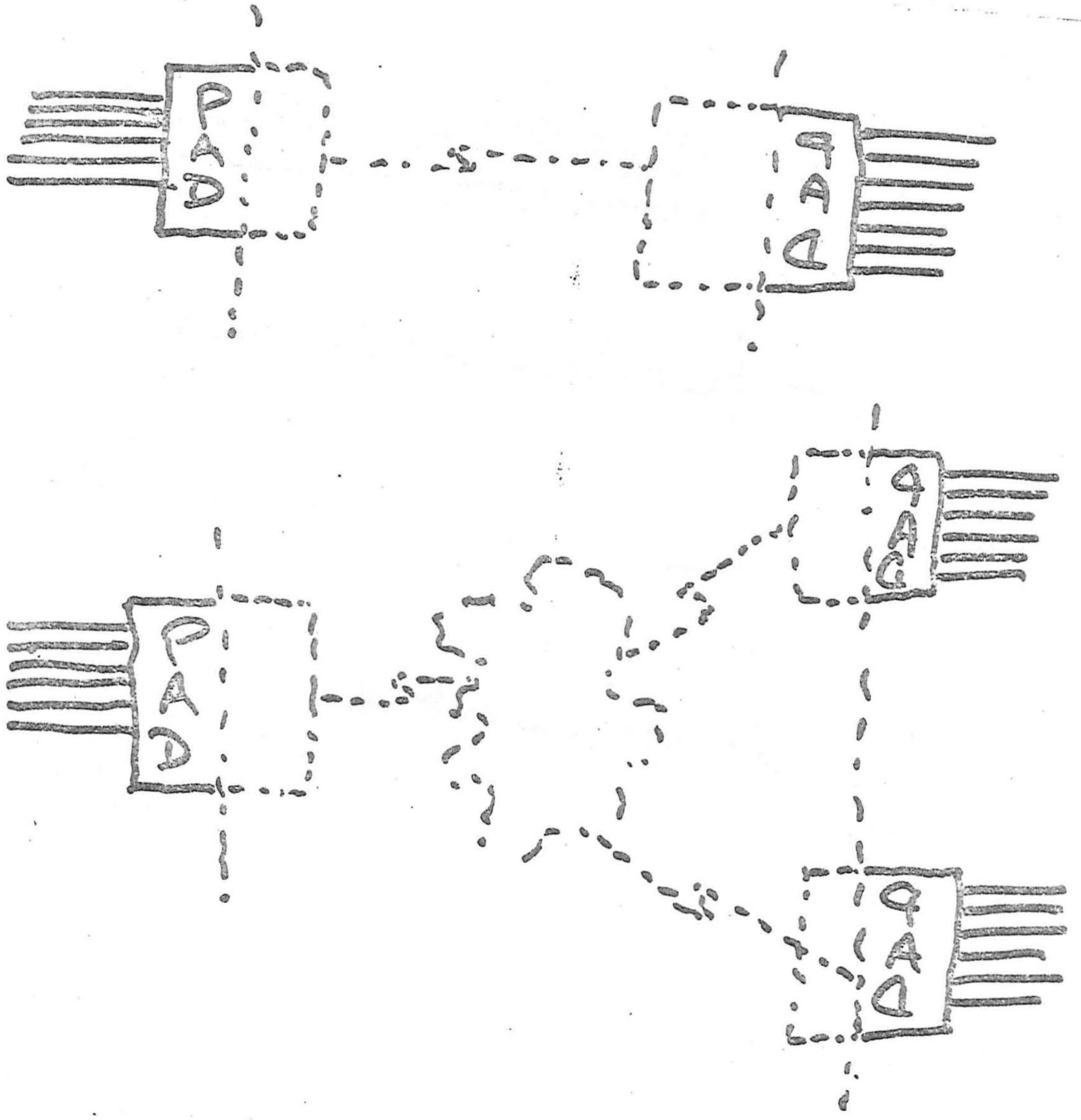
- SIGNIFICANT GROWTH IN TERMINAL USAGE

- SWITCHING REQUIREMENT

- OFF-SITE ACCESS

LOGICAL FUNCTIONS

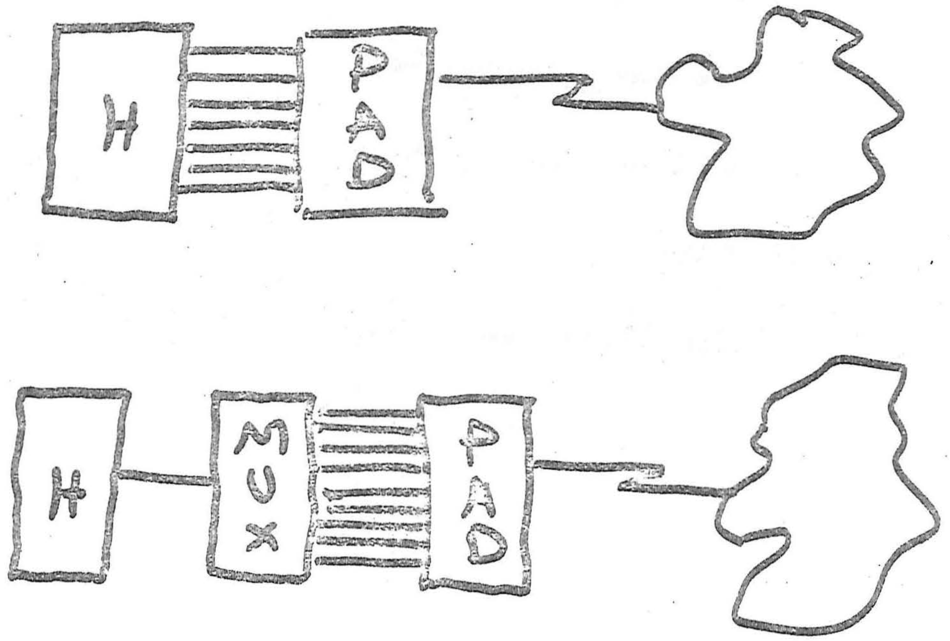
- PAD
- REVERSE PAD (9AO)
- CONCENTRATOR
- SWITCHING / CONTENTION


PAD
PAD
PAD
PAD
PAD

PROGRAMMABILITY

- flexible (reprogrammable)
- maintainable (? by manufacturer)
- expandable

DAP



PROBLEM AREAS FOR REVERSE PAD

- Make + break of connections (to or by host)
- Echoing
- Forwarding conditions
- character mapping
- Parity
- simultaneous input/output

REQUIREMENTS FOR X25 PAD

- Speaks xxx (+TS29)
- Retains known NUIs and mnemonic addresses
- Records call statistics
- Monitors line status
- Concentrates variable #s of terminals
- Polices usage
- Port enable/disable facility
- High availability
- Low cost per port

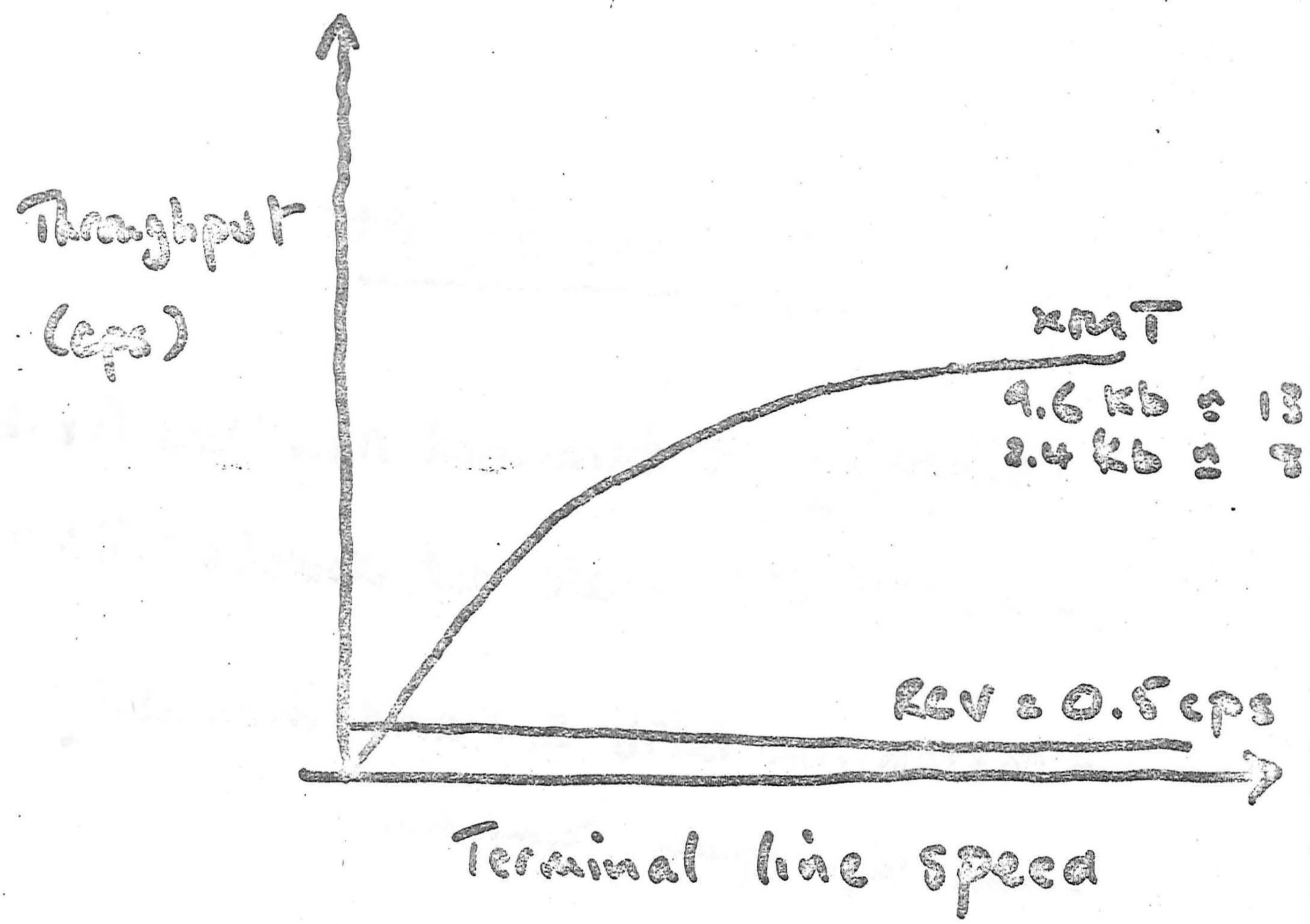
WHAT TYPE OF DEVICE?

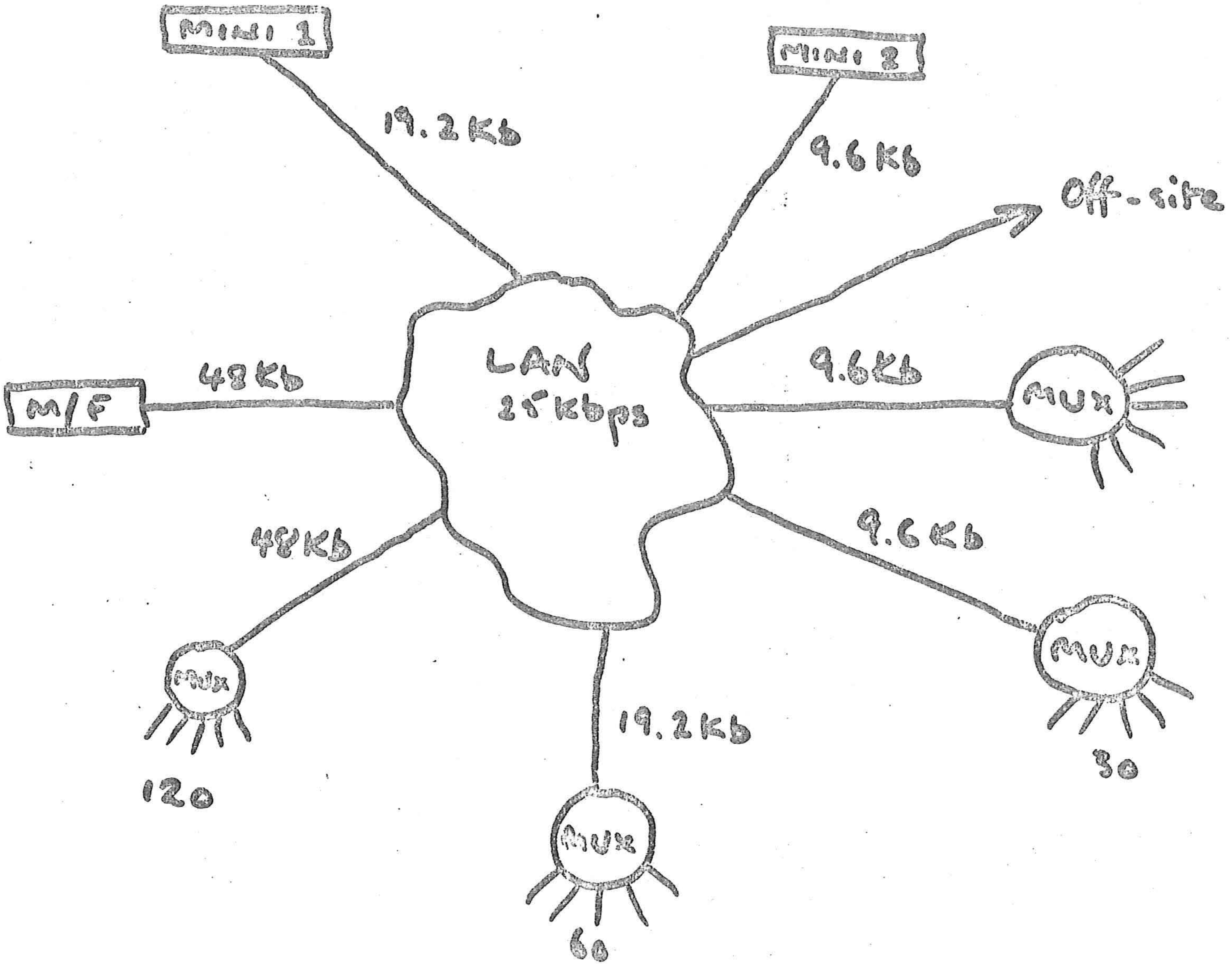
- micro/mini
- Multiple terminal interfaces
- one or more sync lines
- floppy/hard disks for
  - logging
  - NUI store
- expandability

PROJECTIONS FOR 1985

- doubling of terminal numbers (120-240)
- active terminals not doubled (65-115)
- Contention ratio 2:1 and increasing
- more high speed terminals

TERMINAL THROUGHPUT




MINI 1
MINI 2
19.2Kb
9.6Kb
Off-site
M/F
48Kb
LAN
25Kbps
9.6Kb
MUX
48Kb
MUX
120
19.2Kb
MUX
60
9.6Kb
MUX
30

CAMPUS X25 SWITCHES : A SURVEY

Ken Heard

Joint Network Team

CAMPUS X25 SWITCHES

The Networkshop programme anticipated that John Thomas (SWURCC) would be giving this presentation. He is unfortunately unable to be with us, so I will do my best to convey our present understanding of both the potential use and product sources for X25 based switches.

Requirements for a Switch

An X25 based switch may be used to satisfy a number of rather distinct needs:

- connection of several intelligent devices (such as RJE stations, data collecting/editing minis, etc) to a central mainframe, thus acting as a concentrator.
- interconnection of various geographically separated machines, of comparable power, to allow resource sharing through interworking. This is currently of secondary importance but is likely to grow.
- connection of simple, unintelligent devices, such as keyboard terminals, to a small number (usually one, sometimes two) of mainframes. This is properly a separate PAD function, feeding into a pure switch capability, but topology and economics may suggest that the PAD is integrated within the switch itself.
- provision of a gateway to allow controlled access for all local systems to other off-campus systems, via some external communication system such as PSS or dedicated leased lines.
  This gateway function can of course be provided in one of a number of ways: within an existing mainframe, in the switch itself or as an independent gateway machine interposed between the switch and the external network.

In general, most sites will wish to make use of each of these features sooner or later, and so we are looking to systems which could provide each of these features at an economic cost. Considerations of modularity, performance, reliability and maintenance aspects are, of course, equally important. Ideally, we would like to gain experience with a simple, low cost system that can be progressively (and easily) enhanced to match increased demand for both increased raw traffic and improved facilities. The question is: who can provide the goods?

Market Survey

We have looked at a wide range of potential switch suppliers - something like 20 altogether. The majority of offerings would be based on a standard product-line mini, but most potential suppliers are still at the X25 software development stage. Some are only just talking about getting started. All lack solid experience of real X25 working, at least in the UK. We can however identify two or three of the most promising offers. It happens that these cover very different potential needs, in terms of performance, and unfortunately no single candidate appears able to cover the complete range foreseen for a general campus system.

Looking first at the simplest, and cheapest, possibility. Memotec produce a "black box" based on a Z80 micro which is expected to handle several 9.6K bps lines up to an overall throughput limit of about 60K bps. This device was built for use with the Canadian Datapac network and is marketed in the UK by Nolton Communications. A separate box, based on the same architecture, provides a PAD capability. It is intended that both devices should be PSS compatible, but many questions remain unanswered. Whilst several Memotec switches and PADs can be coupled together to provide enhanced performance, there is very little in the way of network management facilities, and any gateway function has yet to be developed

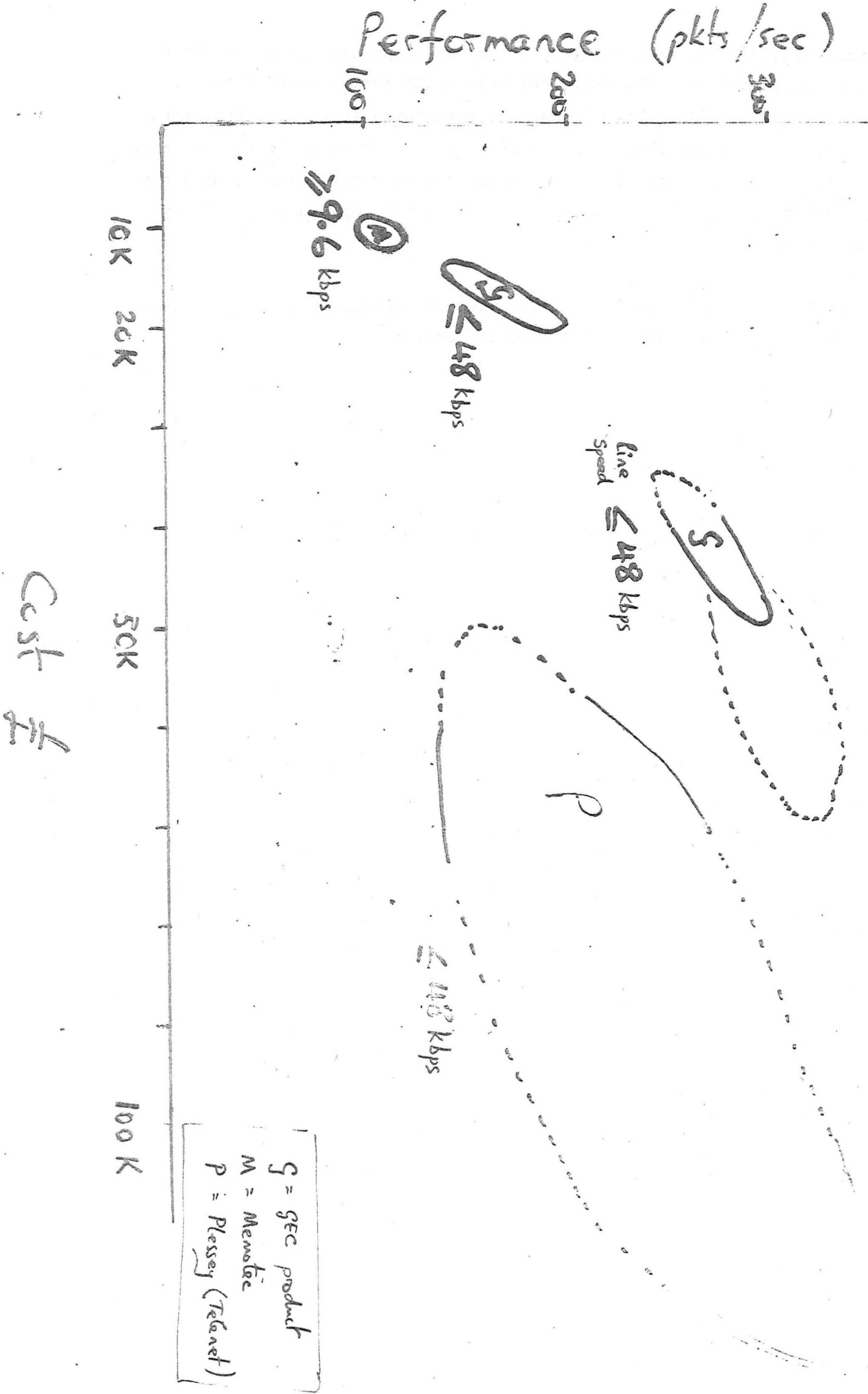
At the other end of the price/performance spectrum, Plessey are marketing the Telenet system based on use of multi-micro TP4000 systems. By definition, this system will be PSS compatible, since it is this kit which is being supplied to the Post Office. It is a highly modular system, which allows any combination of switching and PAD functions to be provided. It is designed around a fully redundant architecture, potentially provides high performance, and is (consequently) rather expensive. Network management features are built into the system and could be controlled and interrogated from a Plessey site over PSS, if we did not want to buy a complete management centre (based on a Prime mini) for an additional £120K!

A third possibility is provided by GEC, based on one of their standard GEC 4000 minicomputers. This system is able to handle line speeds up to 48K bps with an expected packet throughput in the range 150 - 350 packets/second. Switching software is in an advanced state of development, but PAD handling has yet to be provided. A reasonably comprehensive network control and management system has been designed, but this too has yet to be implemented. It is difficult to see how the GEC offering could be developed into a fully redundant, reliable system, based as it is on standard

minicomputers, but then that may not be too important in the context of a single campus. The costs for such a GEC system look to be between £20K - £60K. The GEC system looks quite attractive from both an entry system functionality and cost point of view and there is scope for enhancing its capability over a reasonable range. But it does not yet exist as a product, although there is every hope that it will by the end of the year.

It will anyway take until then for sites to provide an X25 connection capability on the systems they hope to connect.

KSH
18 July 1980


Performance (pkts/sec)
100
200
300
400
Cost £
10K
20K
50K
100K
≥ 9.6 kbps
M
G
≤ 48 kbps
line speed
G
≤ 48 kbps
P
kbps
G = GEC product
M = Memotec
P = Plessey (Telenet)

CONCLUDING REMARKS

Roland Rosner

Joint Network Team

CONCLUDING REMARKS

There were several references during the workshop to development aids, reference centres and certification services for implementations of protocols at all levels. In addition to facilities which may eventually be provided by the Post Office (for X25) and by organisations such as BSI/NCC/NPL (for higher levels), there may be a need for some interim or more permanent arrangements within the academic community. This topic requires further detailed exploration.

Congratulations are due to the JTMP group working under the auspices of the DCPU for producing the draft Job Transfer and Manipulation Protocol in time for this gathering. A period for digestion and discussion now follows and it is hoped that a final draft will be available for the next networkshop.

Following a request at the last networkshop, a contribution was given on tools for programming networking applications. It was concluded that the use of high level languages (such as BCPL) for this work was feasible and that the code could be made as efficient as assembly language implementations. A suggestion was made that a catalogue be kept of cross-compilers, link loaders and associated aids. Since protocols interact heavily with their operating system environments, there was not felt to be much advantage in striving excessively to achieve portability of protocol core algorithms which might only constitute a small part of any given implementation.

A stimulating presentation on user images served to remind us that, if networks are to achieve their objective, considerable thought must be given to making computing resources easily accessible to remote users. A strong plea was entered for the development of context-dependent HELP facilities and the JNT will be pursuing this.

Networking facilities for particular machine ranges were discussed in parallel sessions of which condensed summaries follow.

IBM

All sites run their own flavours of operating systems (OS/MVT, VM, MTS) and their networking plans are accordingly somewhat individualistic. A common intention is the continued use of HASP and there may be some mileage to be gained in trying to create a general package for converting between JTMP and HASP.

ICL

The Dataskil project on 1900's is proceeding well and the goal is to provide (within GEORGE 3) X25, XXX, the Transport Service and a private PAD facility (so that directly connected terminals can access an external network) by Spring 1981. Liverpool and Salford will be the pilot sites and also hope to contribute towards the development of FTP and JTMP respectively. UMRCC are concerned at the implications of further modifications to the operating system. They are accordingly using a GEC 4000 as a front end and implementing built-out networking functions. The timescale is the end of 1980.

No information can yet be published on networking facilities on ICL 2900's. The Computer Board is funding 3 men for 3 years in a collaborative venture with ICL. As an interim measure, QMC are installing micro-based devices to concentrate terminal traffic from X25 networks onto a single physical link carrying ICL protocols.

DEC

There was not sufficient time for detailed discussion on each of the machine range/operating systems, but a summary of the present position was arrived at. Projects are in hand to produce (within a year) the implementations in the table below:-

	X25	XXX	TS	FTP
DEC 10 (TOPS 10)	√	√	√	√
PDP 11 (UNIX)	√	√	√	√
PDP 11 LSI 11 (RT-11)	√		√	√
PDP 11 (RSX-11-M)	√	√		
VAX (VMS)	√	√		

120

It may prove easy to modify the PDP-11 UNIX products for operation under VAX UNIX. The position of networking facilities for DEC 20 (TOPS 20) is still unclear with several options to be explored.

Prime

The company are ready to connect to PSS though modifications are needed to handle some aspects of the service. Plans are being made to implement the Transport Service and FTP and customers will wish to know how much these are likely to cost. Prime urgently need reference sites against which to test their products. They intend to release the definitions of their Fortran interface to the Transport Service and the user interface to FTP. Salford may take on the task of implementing the JTMP.

Honeywell

Information from the company is difficult to obtain. CII-HB in France have a remit to produce networking facilities though there are reservations about their rumoured intentions.

GEC were unfortunately unable to be represented at the workshop.

Local area networks are generating considerable interest in the community though the presentations at the workshop showed that many issues remain to be resolved. Firstly, appropriate components are not yet available off-the-shelf at reasonable cost. Under these circumstances, it is inconceivable that the Computer Board could afford to install production networks at all those universities who may feel they have a case. Secondly, the true costs of local networks must include the connection of subscriber devices and the implementation on them of standard protocols.

Roland Rosner
Joint Network Team

121

PROGRAMME FOR NETWORKSHOP 6

The programme is divided into main sessions to which all are welcome and smaller "birds of a feather" group discussions.

The sessions and times listed below should be treated as provisional.

WEDNESDAY 9 APRIL

Small discussion groups

1700 High-level protocol testing and certification
- leader: Keith Bartlett

1800 Transport Service Implementors
- leader: Keith Bartlett

122

THURSDAY 10 APRIL

Session 1: Introduction and PSS   Chairman: Tony Young

0900 Welcome -

0905 JNT report - Roland Rosner (JNT)

0925 Status report on PSS and connection to IPSS and Euronet - Pat Morrison (PO)

0945 Status reports from early PSS subscribers -
Mike Guy (Cambridge), Ian Service (York)

1000 Discussion

1015 Coffee

Session 2: Protocols   Chairman : Ken Heard

1045 Transport Service status report - Peter Linington (DCPU)

1100 Job Transfer and Manipulation Protocol - Robin John (Dataskil)

1130 TS29/XXX status report - Peter Higginson (UCL)

1145 File Transfer Protocol status report - Dave Rayner (NPL)

1200 Discussion

1215 Lunch

Session 3: General Topics   Chairman: Roland Rosner

1330 High-Level Languages and Compilers for Network Implementation -
John Davies (ULCC)

1415 Planning the transition to standard networking in London -
Tony Peatfield (ULCC)

1445 User's view of accessing services over networks -
Adrian Stokes (Hatfield)

1530 Tea

Small discussion groups

1600 PO Development Aid

- leader: Ian Service

1700 Machine range parallel sessions:

IBM	- leader:	Roland Rosner
ICL	- leader:	Bob Cooper
DEC	- leader:	Barrie Charles
HONEYWELL	- leader:	Ken Heard

123

Friday 11 April

Session 4: Operation and Management   Chairman: Bob Cooper

0900 Network Operation and Management in wide-area networks - Paul Kummer (Daresbury)

0945 Providing and Managing a local network service - Ian Dallas (Kent)

1015 Discussion

1030 Coffee

Session 5: Local Area Networks   Chairman: Barrie Charles

1045 Local Area Networks: a summary report - Ken Heard (JNT)

1115 Interim and Longer Term solutions for terminal handling - Rick Blake (Essex)

1145 Campus X25 Switches: a survey - Ken Heard (JNT)

1215 Summary and General Discussion

1245 Lunch

124

Delegates by Name

Amos,Mr Chris	Birmingham University
Anderson,Mr Wallace	Aberdeen University
Ashton,Mr Ian	Bradford University
Aspden,Mr John	Newcastle University
Bal,Mr G.S.	Liverpool University
Barkataki,Dr Shan	Teesside Polytechnic
Barry,Mr Peter	Glasgow University
Bartlett,Mr Keith	Data Communications Protocol Unit
Blake,Dr Rick	Essex University
Bonney,Mr Norman	Brunel University
Boodson,Mr Alan	Leeds Polytechnic
Bradshaw,Mr Rob	Avon Universities Computer Centre
Brandon,Mr Jeremy	Queen Mary College
Bryant,Dr Paul	Rutherford Laboratory
Burren,Mr John	Rutherford Laboratory
Bye,Mr Colin	Dundee University
Chambers,Mr Alan	Bristol University
Charles,Dr Barrie	Joint Network Team
Chisholm,Mr Ray	Edinburgh Regional Computing Centre
Clark,Mr Tim	Warwick University
Cooper,Dr Bob	Joint Network Team
Cooper,Dr Chris	Rutherford Laboratory
Cooper,Mr Norman	Sussex University
Couzens,Mr Jonathan	Imperial College
Craig,Mr Les	Glasgow University
Dalby,Mr David	London School of Economics
Dallas,Mr Ian	Kent University
Davies,Dr Howard	SWUCN

Davies,Mr John	ULCC
Diecw,Mr Alec	Bradford University
Early,Mr Gordon	Portsmouth Polytechnic
Ellison,Mr David	Salford University
Findon,Mr P.J.	Aston University
Finlayson,Ms Julia	Heriot-Watt University
Foster,Mr John	UMRCC
Foster,Mrs Jill	Newcastle University
Girard,Mr Peter	Rutherford Laboratory
Greenwood,Mr Lyn	Newcastle University
Guy,Mr Mike	Cambridge University
Harrison,Mr Phil	Nottingham University
Hay,Dr Bill	Edinburgh Regional Computing Centre
Heard,Dr Ken	Joint Network Team
Herdman,Mr David	Open University
Higginson,Mr Peter	University College London
Holt,Mr Andy	City University
Hughes,Mr David	Natural Environment Research Council
Hulley,Miss Elaine	Cranfield Institute of Technology
Jack,Mr Andy	Nottingham University
Jamieson,Mr Jim	Strathclyde University
Jennings,Dr David	UWIST
John,Dr Robin	ICL Dataskil
Johnson,Mr Derek	UMRCC
Johnson,Mr Mike	Westfield College
Jones,Dr Paul	Durham University
Kennington,Mr Chris	University College London
Ketteringham,Mr Phil	ICL Dataskil
Kummer,Dr Paul	Daresbury Laboratory
Larmouth,Dr John	Salford University
Lee,Mr David	Durham University

Linington,Dr Peter	Data Communications Protocol Unit
Linn,Dr John	Aberdeen University
Marshall,Mrs Barbara	Southampton University
Morris,Mr Jim	Leicester University
Morrison,Mr Pat	Post Office
Oakley,Mr Mike	Cambridge University
Peatfield,Mr Tony	ULCC
Piotrowicz,Mr Mike	St.Andrews University
Potter,Mr Ben	ICL Dataskil
Powell,Mr Kit	Avon Universities Computer Centre
Rayner,Dr Dave	National Physical Laboratory
Rischmiller,Mr David	Oxford University
Roberts,Miss Sheila	Bristol University
Rosner,Dr Roland	Joint Network Team
Russell,Dr Denis	Newcastle University
Ruttle,Mr Keith	York University
Sage,Mr Morley	Southampton University
Salter,Mr John	ICL Dataskil
Sayers,Dr Mike	Hatfield Polytechnic
Service,Mr Ian	York University
Sheil,Mr Peter	Middlesex Polytechnic
Shelvey,Mr Roger	DEC
Spridgeon,Mr Dave	Hull University
Stephens,G.P.	SWURCC
Stokes,Mr Adrian	Hatfield Polytechnic
Swindells,Mr W	UMIST
Tasker,Dr Robin	Daresbury Laboratory
Thirlby,Mr Rob	Loughborough University of Technology
Thompson,Mr H.E.	University College Cardiff
Tucker,Mr Ken	UWIST
Vivian,Mr Richard	Birmingham University

Wallace,Mr T.L.	PRIME Computer
Walters,Mr C	Brighton Polytechnic
Wanless,Dr Derek	Keele University
Ward,Mr Roger	Bradford University
Wilkinson,Mr K	National Physical Laboratory
Williams,Mr A.H.	Bristol University
Young,Mr Tony	Durham University

Delegates by Institution

Aberdeen University

Anderson,Mr Wallace
Linn,Dr John

Aston University

Findon,Mr P.J.

Avon Universities Computer Centre

Bradshaw,Mr Rob
Powell,Mr Kit

Birmingham University

Amos,Mr Chris
Vivian,Mr Richard

Bradford University

Ashton,Mr Ian
Diedw,Mr Alec
Ward,Mr Roger

Brighton Polytechnic

Walters,Mr C

Bristol University

Chambers,Mr Alan
Roberts,Miss Sheila
Williams,Mr A.H.

Brunel University

Bonney,Mr Norman

Cambridge University

Guy,Mr Mike
Oakley,Mr Mike

City University

Holt,Mr Andy

Cranfield Institute of Technology

Hulley,Miss Elaine

Daresbury Laboratory	
	Kummer,Dr Paul Tasker,Dr Robin
Data Communications Protocol Unit	
	Bartlett,Mr Keith Linington,Dr Peter
DEC	
	Shelvey,Mr Roger
Dundee University	
	Bye,Mr Colin
Durham University	
	Jones,Dr Paul Lee,Mr David Young,Mr Tony
Edinburgh Regional Computing Centre	
	Chisholm,Mr Ray Hay,Dr Bill
Essex University	
	Blake,Dr Rick
Glasgow University	
	Barry,Mr Peter Craig,Mr Les
Hatfield Polytechnic	
	Stokes,Mr Adrian Sayers,Dr Mike
Heriot-Watt University	
	Finlayson,Ms Julia
Hull University	
	Spridgeon,Mr Dave
ICL Dataskil	
	John,Dr Robin Ketteringham,Mr Phil Potter,Mr Ben Salter,Mr John

Imperial College	
	Couzens,Mr Jonathan
Joint Network Team	
	Charles,Dr Barrie Cooper,Dr Bob Heard,Dr Ken Rosner,Dr Roland
Keele University	
	Wanless,Dr Derek
Kent University	
	Dallas,Mr Ian
Leeds Polytechnic	
	Boodson,Mr Alan
Leicester University	
	Morris,Mr Jim
Liverpool University	
	Bal,Mr G.S.
London School of Economics	
	Dalby,Mr David
Loughborough University of Technology	
	Thirlby,Mr Rob
Middlesex Polytechnic	
	Sheil,Mr Peter
National Physical Laboratory	
	Rayner,Dr Dave Wilkinson,Mr K
Natural Environment Research Council	
	Hughes,Mr David
Newcastle University	
	Aspden,Mr John Foster,Mrs Jill Greenwood,Mr Lyn Russell,Dr Denis

Nottingham University	
	Harrison,Mr Phil Jack,Mr Andy
Open University	
	Herdman,Mr David
Oxford University	
	Rischmiller,Mr David
Portsmouth Polytechnic	
	Early,Mr Gordon
Post Office	
	Morrison,Mr Pat
PRIME Computer	
	Wallace,Mr T.L.
Queen Mary College	
	Brandon,Mr Jeremy
Rutherford Laboratory	
	Bryant,Dr Paul Burren,Mr John Cooper,Dr Chris Girard,Mr Peter
Salford University	
	Ellison,Mr David Larmouth,Dr John
Southampton University	
	Marshall,Mrs Barbara Sage,Mr Morley
St.Andrews University	
	Piotrowicz,Mr Mike
Strathclyde University	
	Jamieson,Mr Jim
Sussex University	
	Cooper,Mr Norman

SWUCN	
	Davies,Dr Howard
SWURCC	
	Stephens,G.P.
Teesside Polytechnic	
	Barkataki,Dr Shan
ULCC	
	Davies,Mr John Peatfield,Mr Tony
UMIST	
	Swindells,Mr W
UMRCC	
	Foster,Mr John Johnson,Mr Derek
University College Cardiff	
	Thompson,Mr H.E.
University College London	
	Higginson,Mr Peter Kennington,Mr Chris
UWIST	
	Jennings,Dr David Tucker,Mr Ken
Warwick University	
	Clark,Mr Tim
Westfield College	
	Johnson,Mr Mike
York University	
	Service,Mr Ian Ruttle,Mr Keith